Siren User Guide

Release 6.0.8

Esther Galbrun and Pauli Miettinen

November, 2021

CONTENTS

1 Getting Started 3

2 Interface 5

3 Formats 23

4 Parameters 27

Index 55

1. Prepare and load your data

2. Launch an algorithm to mine redescriptions

3. Visualize, filter, edit and save the redescriptions

Iterate…

CHAPTER

ONE

GETTING STARTED

Note: *Siren* allows you to interactively mine and visualize redescriptions from your data.

We outline here some high-level interactions offered by *Siren*.

Because examples are worth many words, here are some datasets to try your hands on.

1.1 1. Load data

After you get *Siren* installed and running, it will open the *tools window*, with tabs for variables and redescriptions, all of them empty. Hence, the first thing to do is to import data or open an existing siren package to start working.

If you already have some siren package (i.e. with a `.siren` extension) you can open it via the interface menu *File* → *Open*.

Otherwise, or if you want to work on a new data set, you can *import data* to *Siren*, this will populate the `Entities` and `Variables` tabs.

1.2 2. Mine redescriptions

Once the two sets of variables are loaded and can be seen in the tabs you may want to let the tool mine redescriptions in an fully automated way, using currently enabled variables. The menu entry *Process* → *Mine* redescriptions let you do just that.

Alternatively, if both queries in a visualization are empty *Siren* simply mines redescriptions on your data when clicking on the `Expand` button.

Results of the mining process will appear in a new list in the `Redescriptions` tab.

Before running a mining task make sure you have adjusted the main *mining parameters*...

1.3 3. Visualize and edit redescriptions

1.3.1 Visualizing a redescription

Individual redescriptions and lists can be plotted in different visualizations, depending on the type of your data and the redescriptions.

Double-click on an item (variable, redescription, list) to open the default visualization for that item. The *View* entry in the main menu and under the contextual menu list available visualizations.

1.3.2 Expanding a redescription

You can also press the expansion button to automatically find expansions of any redescription your are currently visualizing and editing. *Siren* will try to append literals to the current redescription, using the enabled variables.

Again, results will be appended to the redescriptions list in the `Expansions` tab.

Before running an expansion task make sure you have adjusted the *mining parameters*…

1.3.3 Filtering redescriptions

A list of redescription can be filtered automatically. That is, the algorithm will go through the redescriptions, from top to bottom in the current order, an check for each redescription, whether it is redundant given the previous ones. If a redescription is found redundant it will be disabled. This is done via the different filter entries in the *Edit* menu .

1.3.4 Exporting redescriptions

The lists of redescriptions from the *Redescriptions* tab can be exported under *different formats* from the contextual menu.

1.3.5 Saving as a package

You can save your current project, i.e. the data, current redescription lists and parameter preferences as a siren package (i.e. with a `.siren` extension) via the interface menu *Edit* → *Save as…*

If you continue working on the current project you can save changes to the current siren package via the interface menu *Edit* → *Save*.

Existing siren packages can be opened via the interface menu *File* → *Open*.

Lists of redescriptions can be added and deleted from a package. A package needs to be first created before redescriptions lists can be added to it.

1.4 Iterate…

The data analysis process typically require a few iterations of mining, looking at the results, tuning the parameters, etc.

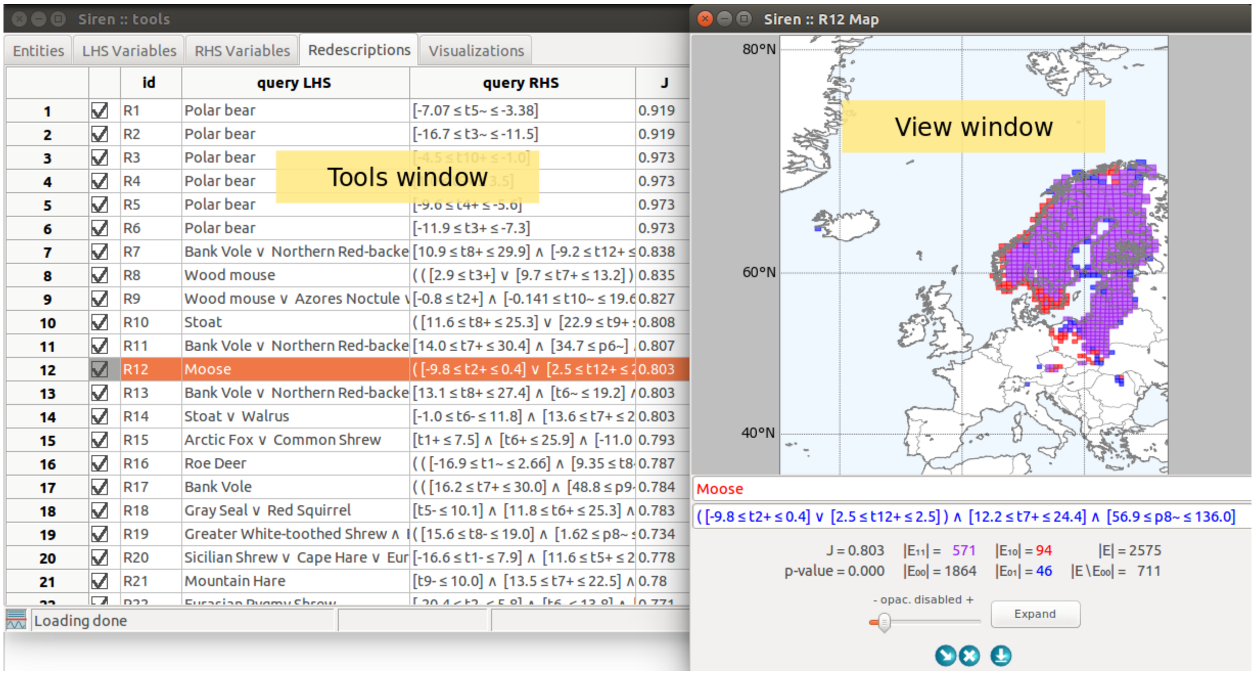
CHAPTER

TWO

INTERFACE

2.1 Windows and Tabs

Note: *Siren* has two types of windows: *tools* and *view*, which are presented below.



		id	query LHS	query RHS	J
1	☑	R1	Polar bear	[-7.07 ≤ t5~ ≤ -3.38]	0.919
2	☑	R2	Polar bear	[-16.7 ≤ t3~ ≤ -11.5]	0.919
3	☑	R3	Polar bear	[-4.5 ≤ t10+ ≤ -1.0]	0.973
4	☑	R4	Polar bear	[illegible]	0.973
5	☑	R5	Polar bear	[-9.6 ≤ t4+ ≤ -5.6]	0.973
6	☑	R6	Polar bear	[-11.9 ≤ t3+ ≤ -7.3]	0.973
7	☑	R7	Bank Vole ∨ Northern Red-backe	[10.9 ≤ t8+ ≤ 29.9] ∧ [-9.2 ≤ t12+ ≤	0.838
8	☑	R8	Wood mouse	(([2.9 ≤ t3+] ∨ [9.7 ≤ t7+ ≤ 13.2])	0.835
9	☑	R9	Wood mouse ∨ Azores Noctule ∨	[-0.8 ≤ t2+] ∧ [-0.141 ≤ t10~ ≤ 19.6	0.827
10	☑	R10	Stoat	([11.6 ≤ t8+ ≤ 25.3] ∨ [22.9 ≤ t9+ ≤	0.808
11	☑	R11	Bank Vole ∨ Northern Red-backe	[14.0 ≤ t7+ ≤ 30.4] ∧ [34.7 ≤ p6~] ∧	0.807
12	☑	R12	Moose	([-9.8 ≤ t2+ ≤ 0.4] ∨ [2.5 ≤ t12+ ≤ 2	0.803
13	☑	R13	Bank Vole ∨ Northern Red-backe	[13.1 ≤ t8+ ≤ 27.4] ∧ [t6~ ≤ 19.2] ∧	0.803
14	☑	R14	Stoat ∨ Walrus	[-1.0 ≤ t6- ≤ 11.8] ∧ [13.6 ≤ t7+ ≤ 2	0.803
15	☑	R15	Arctic Fox ∨ Common Shrew	[t1+ ≤ 7.5] ∧ [t6+ ≤ 25.9] ∧ [-11.0	0.793
16	☑	R16	Roe Deer	(([-16.9 ≤ t1~ ≤ 2.66] ∧ [9.35 ≤ t8	0.787
17	☑	R17	Bank Vole	(([16.2 ≤ t7+ ≤ 30.0] ∧ [48.8 ≤ p9	0.784
18	☑	R18	Gray Seal ∨ Red Squirrel	[t5- ≤ 10.1] ∧ [11.8 ≤ t6+ ≤ 25.3] ∧	0.783
19	☑	R19	Greater White-toothed Shrew ∧	([15.6 ≤ t8- ≤ 19.0] ∧ [1.62 ≤ p8~ ≤	0.734
20	☑	R20	Sicilian Shrew ∨ Cape Hare ∨ Eur	[-16.6 ≤ t1- ≤ 7.9] ∧ [11.6 ≤ t5+ ≤ 2	0.778
21	☑	R21	Mountain Hare	[t9- ≤ 10.0] ∧ [13.5 ≤ t7+ ≤ 22.5] ∧	0.78

Loading done

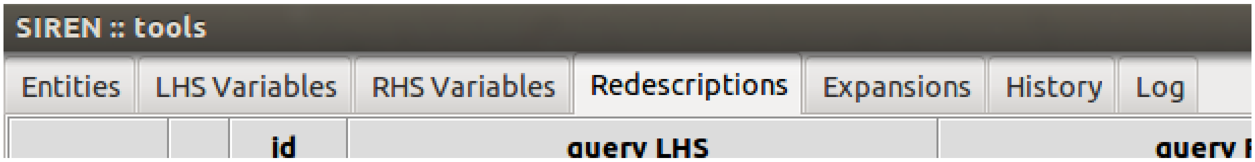
Moose

([-9.8 ≤ t2+ ≤ 0.4] ∨ [2.5 ≤ t12+ ≤ 2.5]) ∧ [12.2 ≤ t7+ ≤ 24.4] ∧ [56.9 ≤ p8~ ≤ 136.0]

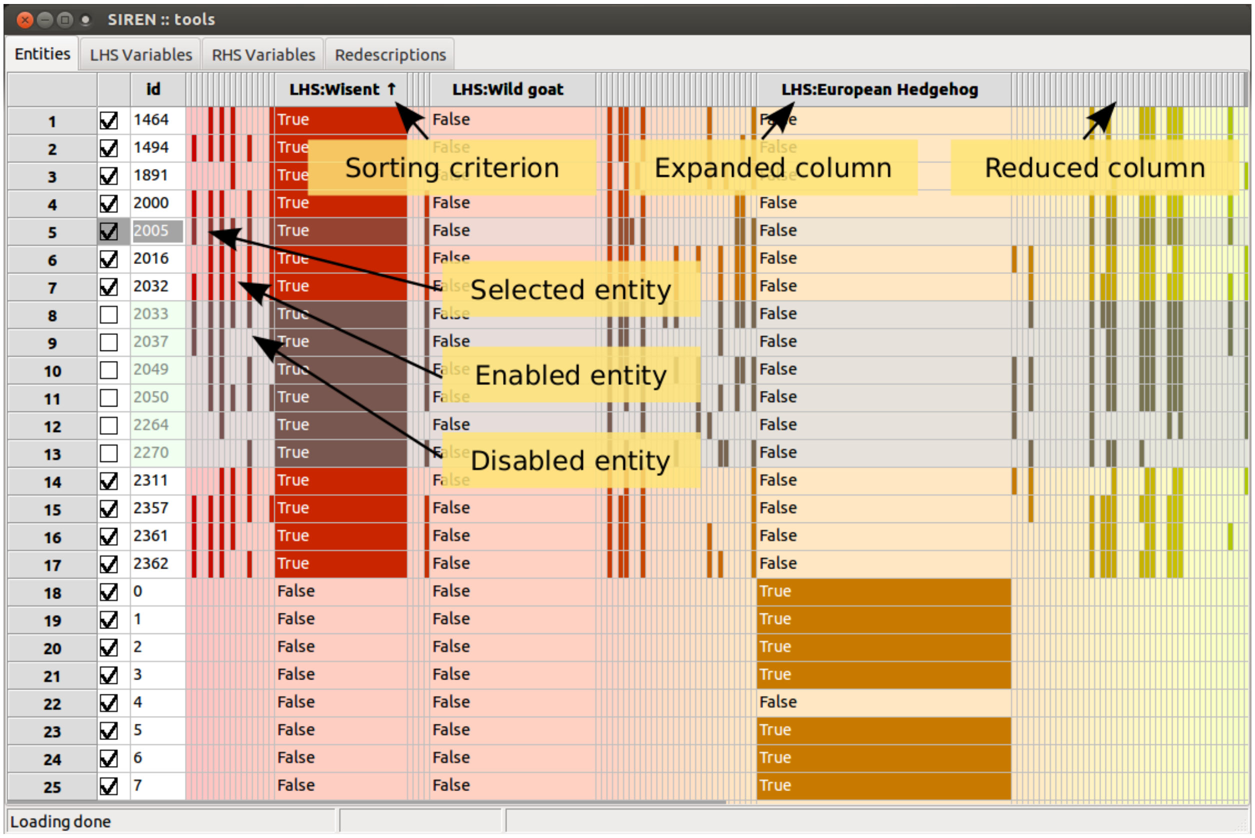
J = 0.803 |E₁₁| = 571 |E₁₀| = 94 |E| = 2575
p-value = 0.000 |E₀₀| = 1864 |E₀₁| = 46 |E \ E₀₀| = 711

2.1.1 Tools window

The `Tools` window is unique, it is the main siren window. It contains several tabs.



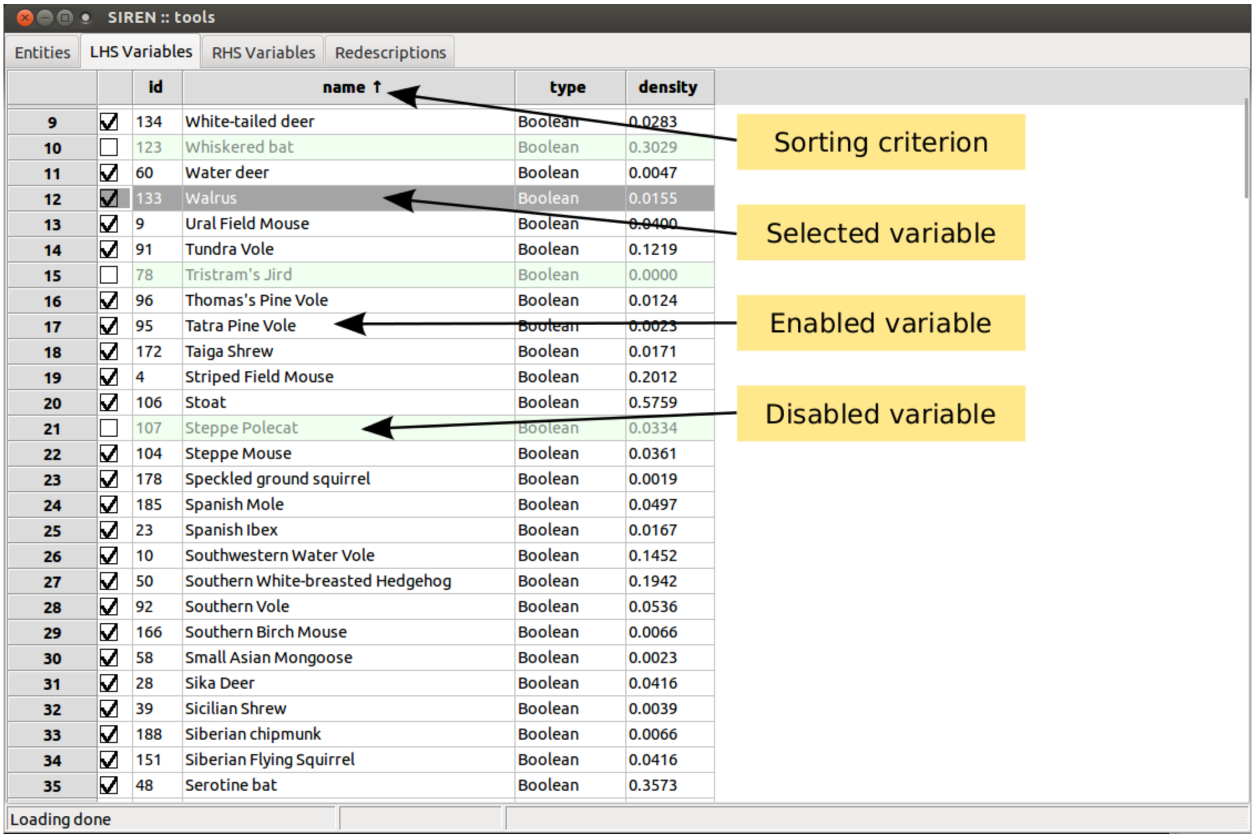
- The **Entities** tab contains the list of entities.



		id	LHS:Wisent ↑	LHS:Wild goat	LHS:European Hedgehog
1	☑	1464	True	False	False
2	☑	1494	True	False	False
3	☑	1891	True	False	False
4	☑	2000	True	False	False
5	☑	2005	True	False	False
6	☑	2016	True	False	False
7	☑	2032	True	False	False
8	☐	2033	True	False	False
9	☐	2037	True	False	False
10	☐	2049	True	False	False
11	☐	2050	True	False	False
12	☐	2264	True	False	False
13	☐	2270	True	False	False
14	☑	2311	True	False	False
15	☑	2357	True	False	False
16	☑	2361	True	False	False
17	☑	2362	True	False	False
18	☑	0	False	False	True
19	☑	1	False	False	True
20	☑	2	False	False	True
21	☑	3	False	False	True
22	☑	4	False	False	False
23	☑	5	False	False	True
24	☑	6	False	False	True
25	☑	7	False	False	True

Two tabs contain lists of variables.

- The **LHS Variables** tab,
- and the **RHS Variables** tab contain the list of left-hand side variables and right-hand side variables respectively.



		id	name ↑	type	density
9	☑	134	White-tailed deer	Boolean	0.0283
10	☐	123	Whiskered bat	Boolean	0.3029
11	☑	60	Water deer	Boolean	0.0047
12	☑	133	Walrus	Boolean	0.0155
13	☑	9	Ural Field Mouse	Boolean	0.0400
14	☑	91	Tundra Vole	Boolean	0.1219
15	☐	78	Tristram's Jird	Boolean	0.0000
16	☑	96	Thomas's Pine Vole	Boolean	0.0124
17	☑	95	Tatra Pine Vole	Boolean	0.0023
18	☑	172	Taiga Shrew	Boolean	0.0171
19	☑	4	Striped Field Mouse	Boolean	0.2012
20	☑	106	Stoat	Boolean	0.5759
21	☐	107	Steppe Polecat	Boolean	0.0334
22	☑	104	Steppe Mouse	Boolean	0.0361
23	☑	178	Speckled ground squirrel	Boolean	0.0019
24	☑	185	Spanish Mole	Boolean	0.0497
25	☑	23	Spanish Ibex	Boolean	0.0167
26	☑	10	Southwestern Water Vole	Boolean	0.1452
27	☑	50	Southern White-breasted Hedgehog	Boolean	0.1942
28	☑	92	Southern Vole	Boolean	0.0536
29	☑	166	Southern Birch Mouse	Boolean	0.0066
30	☑	58	Small Asian Mongoose	Boolean	0.0023
31	☑	28	Sika Deer	Boolean	0.0416
32	☑	39	Sicilian Shrew	Boolean	0.0039
33	☑	188	Siberian chipmunk	Boolean	0.0066
34	☑	151	Siberian Flying Squirrel	Boolean	0.0416
35	☑	48	Serotine bat	Boolean	0.3573

Sorting criterion

Selected variable

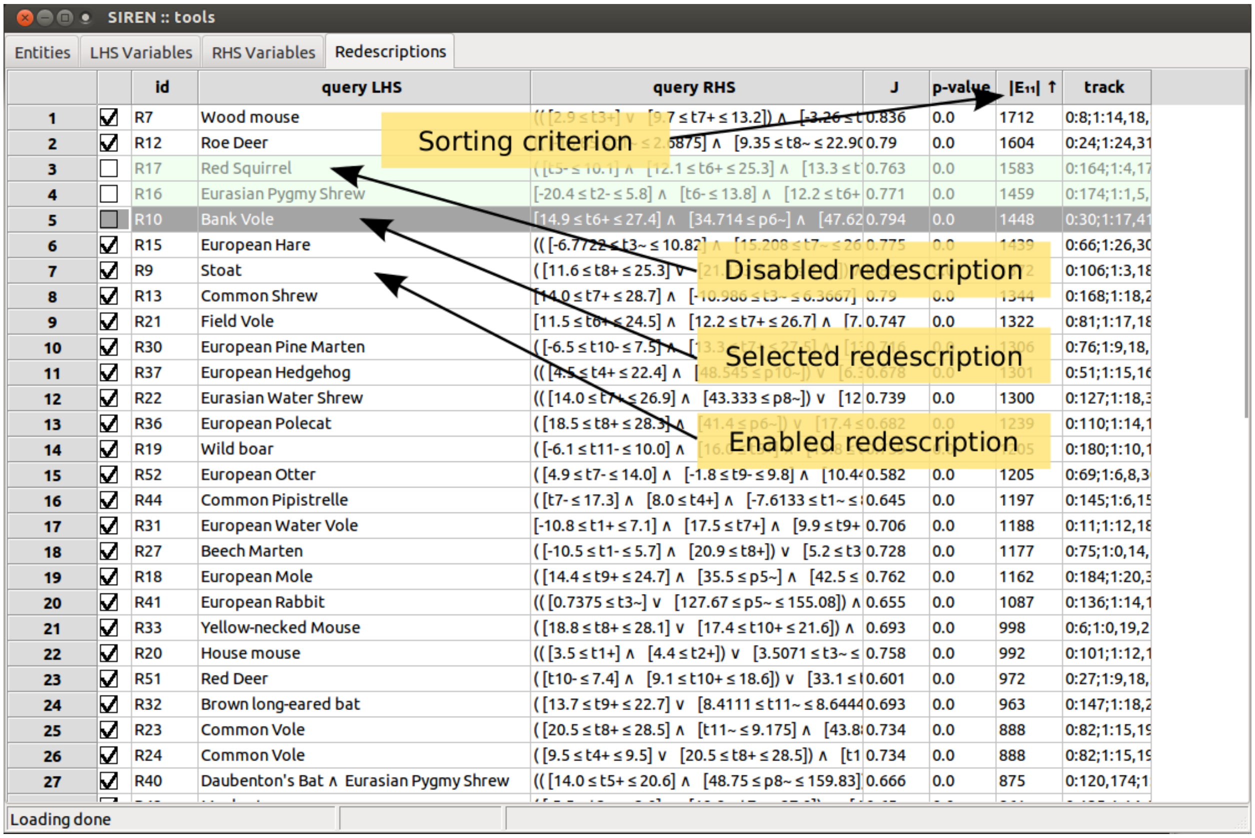
Enabled variable

Disabled variable

Loading done

Three tabs contain lists of redescriptions.

- The **Redescriptions** tab is the main list of redescriptions. Redescriptions are imported and exported to and from that list.
- The **Expansions** tab lists redescriptions that have been generated by mining or extending queries.
- The **History** tab lists all edits made to queries, allowing to undo changes.

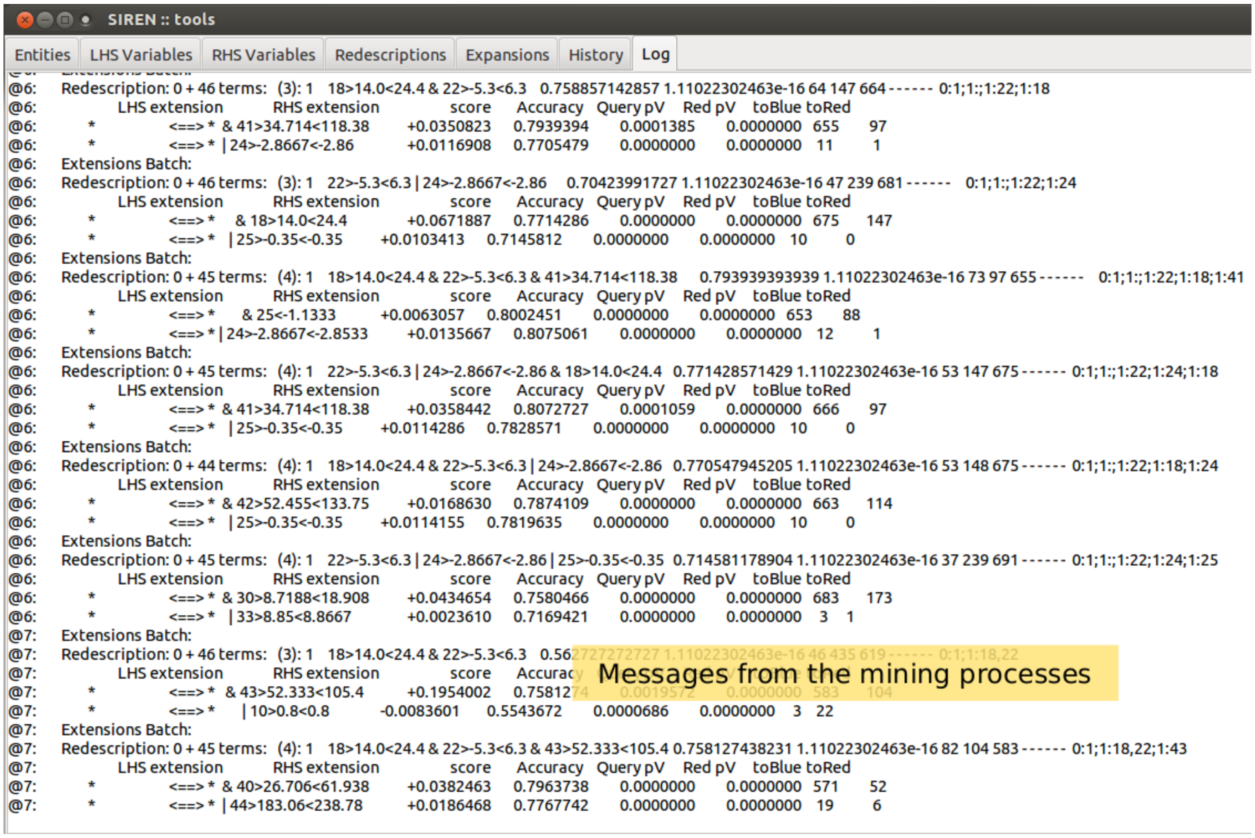


		id	query LHS	query RHS	J	p-value	|E₁₁| ↑	track
1	☑	R7	Wood mouse	(([2.9 ≤ t3+] ∨ [9.7 ≤ t7+ ≤ 13.2]) ∧ [-3.26 ≤ t	0.836	0.0	1712	0:8;1:14,18,
2	☑	R12	Roe Deer	≤ 2.5875] ∧ [9.35 ≤ t8~ ≤ 22.9(	0.79	0.0	1604	0:24;1:24,31
3	☐	R17	Red Squirrel	([t5- ≤ 10.1] ∧ [12.1 ≤ t6+ ≤ 25.3] ∧ [13.3 ≤ t	0.763	0.0	1583	0:164;1:4,17
4	☐	R16	Eurasian Pygmy Shrew	[-20.4 ≤ t2- ≤ 5.8] ∧ [t6- ≤ 13.8] ∧ [12.2 ≤ t6+	0.771	0.0	1459	0:174;1:1,5,
5	■	R10	Bank Vole	[14.9 ≤ t6+ ≤ 27.4] ∧ [34.714 ≤ p6~] ∧ [47.62	0.794	0.0	1448	0:30;1:17,41
6	☑	R15	European Hare	(([-6.7722 ≤ t3~ ≤ 10.82] ∧ [15.208 ≤ t7~ ≤ 26	0.775	0.0	1439	0:66;1:26,30
7	☑	R9	Stoat	([11.6 ≤ t8+ ≤ 25.3] ∨ [21	[illegible]	[illegible]	[illegible]	0:106;1:3,18
8	☑	R13	Common Shrew	[14.0 ≤ t7+ ≤ 28.7] ∧ [-10.986 ≤ t3~ ≤ 6.3667]	0.79	0.0	1344	0:168;1:18,2
9	☑	R21	Field Vole	[11.5 ≤ t6+ ≤ 24.5] ∧ [12.2 ≤ t7+ ≤ 26.7] ∧ [7.	0.747	0.0	1322	0:81;1:17,18
10	☑	R30	European Pine Marten	([-6.5 ≤ t10- ≤ 7.5] ∧ [13.3 ≤ t7~ ≤ 27.5] ∧ [1	0.716	0.0	1306	0:76;1:9,18,
11	☑	R37	European Hedgehog	(([4.5 ≤ t4+ ≤ 22.4] ∧ [48.545 ≤ p10~]) ∨ [6.3	0.678	0.0	1301	0:51;1:15,16
12	☑	R22	Eurasian Water Shrew	(([14.0 ≤ t7+ ≤ 26.9] ∧ [43.333 ≤ p8~]) ∨ [12	0.739	0.0	1300	0:127;1:18,3
13	☑	R36	European Polecat	([18.5 ≤ t8+ ≤ 28.3] ∧ [41.4 ≤ p6~]) ∨ [17.4 ≤	0.682	0.0	1239	0:110;1:14,1
14	☑	R19	Wild boar	([-6.1 ≤ t11- ≤ 10.0] ∧ [16.0	[illegible]	[illegible]	[illegible]	0:180;1:10,1
15	☑	R52	European Otter	([4.9 ≤ t7- ≤ 14.0] ∧ [-1.8 ≤ t9- ≤ 9.8] ∧ [10.4	0.582	0.0	1205	0:69;1:6,8,3
16	☑	R44	Common Pipistrelle	([t7- ≤ 17.3] ∧ [8.0 ≤ t4+] ∧ [-7.6133 ≤ t1~ ≤	0.645	0.0	1197	0:145;1:6,15
17	☑	R31	European Water Vole	[-10.8 ≤ t1+ ≤ 7.1] ∧ [17.5 ≤ t7+] ∧ [9.9 ≤ t9+	0.706	0.0	1188	0:11;1:12,18
18	☑	R27	Beech Marten	([-10.5 ≤ t1- ≤ 5.7] ∧ [20.9 ≤ t8+]) ∨ [5.2 ≤ t3	0.728	0.0	1177	0:75;1:0,14,
19	☑	R18	European Mole	([14.4 ≤ t9+ ≤ 24.7] ∧ [35.5 ≤ p5~] ∧ [42.5 ≤	0.762	0.0	1162	0:184;1:20,3
20	☑	R41	European Rabbit	(([0.7375 ≤ t3~] ∨ [127.67 ≤ p5~ ≤ 155.08]) ∧	0.655	0.0	1087	0:136;1:14,1
21	☑	R33	Yellow-necked Mouse	([18.8 ≤ t8+ ≤ 28.1] ∨ [17.4 ≤ t10+ ≤ 21.6]) ∧	0.693	0.0	998	0:6;1:0,19,2
22	☑	R20	House mouse	(([3.5 ≤ t1+] ∧ [4.4 ≤ t2+]) ∨ [3.5071 ≤ t3~ ≤	0.758	0.0	992	0:101;1:12,1
23	☑	R51	Red Deer	([t10- ≤ 7.4] ∧ [9.1 ≤ t10+ ≤ 18.6]) ∨ [33.1 ≤	0.601	0.0	972	0:27;1:9,18,
24	☑	R32	Brown long-eared bat	([13.7 ≤ t9+ ≤ 22.7] ∨ [8.4111 ≤ t11~ ≤ 8.6444	0.693	0.0	963	0:147;1:18,2
25	☑	R23	Common Vole	([20.5 ≤ t8+ ≤ 28.5] ∧ [t11~ ≤ 9.175] ∧ [43.8	0.734	0.0	888	0:82;1:15,19
26	☑	R24	Common Vole	([9.5 ≤ t4+ ≤ 9.5] ∨ [20.5 ≤ t8+ ≤ 28.5]) ∧ [t1	0.734	0.0	888	0:82;1:15,19
27	☑	R40	Daubenton's Bat ∧ Eurasian Pygmy Shrew	(([14.0 ≤ t5+ ≤ 20.6] ∧ [48.75 ≤ p8~ ≤ 159.83]	0.666	0.0	875	0:120,174;1

Loading done

Finally,

- the **Log** tab contains logging output generated by the mining algorithm.



```
@6:    Redescription: 0 + 46 terms:  (3): 1   18>14.0<24.4 & 22>-5.3<6.3   0.758857142857 1.11022302463e-16 64 147 664 ------ 0:1;1:;1:22;1:18
@6:             LHS extension          RHS extension              score      Accuracy  Query pV    Red pV    toBlue toRed
@6:         *             <==> * & 41>34.714<118.38        +0.0350823    0.7939394      0.0001385       0.0000000  655      97
@6:         *             <==> * | 24>-2.8667<-2.86         +0.0116908    0.7705479      0.0000000       0.0000000  11       1
@6:    Extensions Batch:
@6:    Redescription: 0 + 46 terms:  (3): 1   22>-5.3<6.3 | 24>-2.8667<-2.86    0.70423991727 1.11022302463e-16 47 239 681 ------    0:1;1:;1:22;1:24
@6:             LHS extension          RHS extension              score      Accuracy  Query pV    Red pV    toBlue toRed
@6:         *             <==> *    & 18>14.0<24.4           +0.0671887    0.7714286      0.0000000       0.0000000  675      147
@6:         *             <==> *   | 25>-0.35<-0.35       +0.0103413    0.7145812      0.0000000       0.0000000   10       0
@6:    Extensions Batch:
@6:    Redescription: 0 + 45 terms:  (4): 1   18>14.0<24.4 & 22>-5.3<6.3 & 41>34.714<118.38     0.793939393939 1.11022302463e-16 73 97 655 ------    0:1;1:;1:22;1:18;1:41
@6:             LHS extension          RHS extension              score      Accuracy  Query pV    Red pV    toBlue toRed
@6:         *             <==> *     & 25<-1.1333          +0.0063057    0.8002451      0.0000000       0.0000000  653      88
@6:         *             <==> * | 24>-2.8667<-2.8533       +0.0135667    0.8075061      0.0000000       0.0000000  12       1
@6:    Extensions Batch:
@6:    Redescription: 0 + 45 terms:  (4): 1   22>-5.3<6.3 | 24>-2.8667<-2.86 & 18>14.0<24.4  0.771428571429 1.11022302463e-16 53 147 675 ------ 0:1;1:;1:22;1:24;1:18
@6:             LHS extension          RHS extension              score      Accuracy  Query pV    Red pV    toBlue toRed
@6:         *             <==> * & 41>34.714<118.38        +0.0358442    0.8072727      0.0001059       0.0000000  666      97
@6:         *             <==> *   | 25>-0.35<-0.35       +0.0114286    0.7828571      0.0000000       0.0000000   10       0
@6:    Extensions Batch:
@6:    Redescription: 0 + 44 terms:  (4): 1   18>14.0<24.4 & 22>-5.3<6.3 | 24>-2.8667<-2.86  0.770547945205 1.11022302463e-16 53 148 675 ------ 0:1;1:;1:22;1:18;1:24
@6:             LHS extension          RHS extension              score      Accuracy  Query pV    Red pV    toBlue toRed
@6:         *             <==> * & 42>52.455<133.75        +0.0168630    0.7874109      0.0000000       0.0000000  663      114
@6:         *             <==> *   | 25>-0.35<-0.35       +0.0114155    0.7819635      0.0000000       0.0000000   10       0
@6:    Extensions Batch:
@6:    Redescription: 0 + 45 terms:  (4): 1   22>-5.3<6.3 | 24>-2.8667<-2.86 | 25>-0.35<-0.35  0.714581178904 1.11022302463e-16 37 239 691 ------ 0:1;1:;1:22;1:24;1:25
@6:             LHS extension          RHS extension              score      Accuracy  Query pV    Red pV    toBlue toRed
@6:         *             <==> * & 30>8.7188<18.908        +0.0434654    0.7580466      0.0000000       0.0000000  683      173
@6:         *             <==> *   | 33>8.85<8.8667        +0.0023610    0.7169421      0.0000000       0.0000000   3   1
@7:    Extensions Batch:
@7:    Redescription: 0 + 46 terms:  (3): 1   18>14.0<24.4 & 22>-5.3<6.3   0.562727272727 1.11022302463e-16 46 435 619 ------ 0:1;1:18,22
@7:             LHS extension          RHS extension              score      Accuracy
@7:         *             <==> *  & 43>52.333<105.4        +0.1954002    0.7581274      0.0019572       0.0000000  583      104
@7:         *             <==> *     | 10>0.8<0.8          -0.0083601    0.5543672      0.0000686       0.0000000   3   22
@7:    Extensions Batch:
@7:    Redescription: 0 + 45 terms:  (4): 1   18>14.0<24.4 & 22>-5.3<6.3 & 43>52.333<105.4 0.758127438231 1.11022302463e-16 82 104 583 ------ 0:1;1:18,22;1:43
@7:             LHS extension          RHS extension              score      Accuracy  Query pV    Red pV    toBlue toRed
@7:         *             <==> * & 40>26.706<61.938        +0.0382463    0.7963738      0.0000000       0.0000000  571      52
@7:         *             <==> * | 44>183.06<238.78        +0.0186468    0.7767742      0.0000000       0.0000000  19       6
```

Messages from the mining processes

Tabs can be shown or hidden via the interface menu *Windows* → *Tabs*.

The main means to manipulate variables or redescriptions, depending on which tab you are currently viewing, are available via the `Edit` menu and the contextual menu opened by right click.

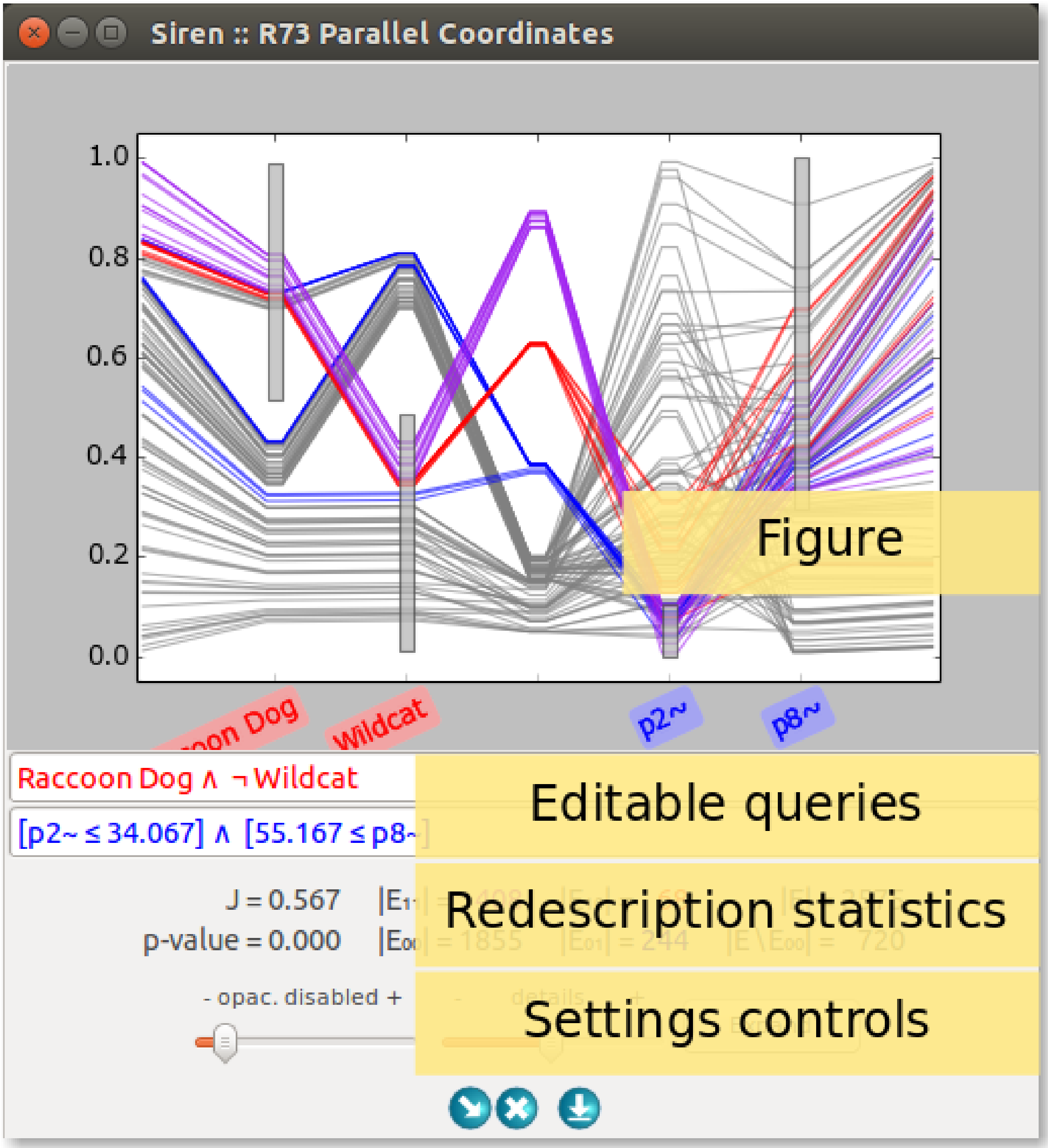
List items can be enabled/disabled by checking/unchecking the corresponding box in the left column. All items in a list can be enabled/disabled simultaneously. Lists can be sorted based on the value of the different fields displayed by clicking on the column header.

Redescriptions can be cut, copied, and pasted from and to different positions in the lists and from one list to the other and vice-versa. All disabled redescriptions can be deleted at once.

2.1.2 View window

A **view** window allows to visualize a redescription, to edit it and launch mining and expansion. Several views of different types of the same redescription can be opened simultaneously. They are linked together an to the original redescription in the list so that edits and selections made in one view are reflected in the list and other views.

A `View` window can be opened with a double click on a variable or redescription in a list from the `Tools` window.



Raccoon Dog ∧ ¬Wildcat

[p2~ ≤ 34.067] ∧ [55.167 ≤ p8~

J = 0.567

p-value = 0.000

Both queries can be edited using the text boxes. If the *syntax of a query* is incorrect, *Siren* will not be able to parse it and it will fall back on the previous correct query. Queries are parsed when `ENTER` is pressed, in order to avoid parsing error due to partial edits.

Redescriptions statistics are shown below the queries.

Expansion can be started by pressing the `Expand` button. The expansion will be delegated to a background process. It can be interrupted via the menu *Process* → *Stop expander XXX*, where running task are listed. Redescriptions generated during the expansion will be appended to the list of redescriptions in the `Expansions` tab.

Clicking on the entities, represented as lines or dots, highlights them accross the different views for the redescription with labels showing the entities's ID to allow identifying them. It is also possible to enable highlighting and labeling the entities on hover (by activating *Hover entities* in the *Interface* preferences), and jumping to the entity in the entities list on click (by toggling *Edit* → *View details*).

Some actions are visualization-specific. A slider allows to set the opacity of disabled entities in all visualizations except the tree diagrams. For parallel coordinates plots, another slider allows to adjust the level of details by limiting the fraction of entities drawn.

In case of projections, setting controls allow to set parameters and the data can be projected anew by clicking the `Reproject` button. A polygon drawing tool allows to select groups of contiguous areas.

In tree diagrams, clicking on a leaf allows to add and remove the corresponding branch to the query. Then, `Simplify LHS` and `Simplify RHS` buttons allow to simplify the diagrams of the tree associated to the left-hand side and right-hand side queries, respectively.

In parallel coordinates plots, the range of literals can be adjusted interactively by dragging the corresponding grey boxes.

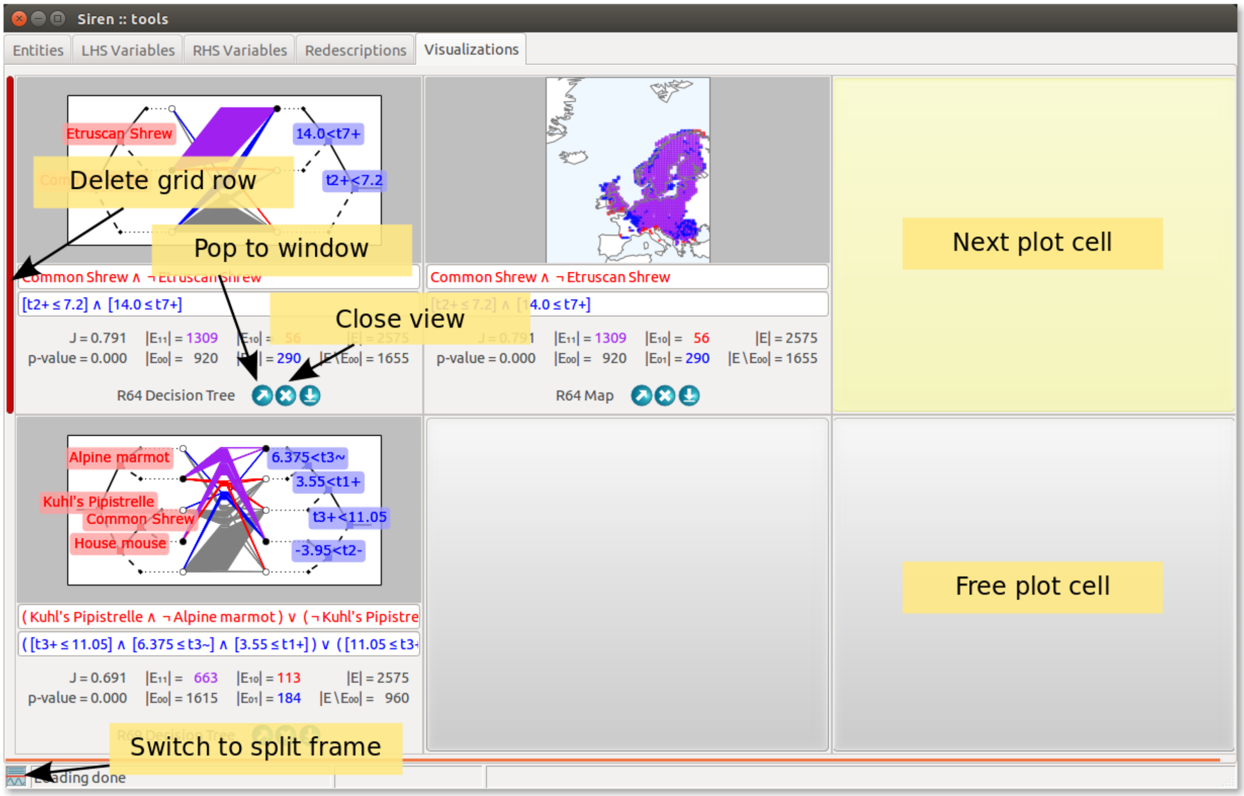
2.1.3 Display modes

Plotting the visualizations to separate windows is one possibility. In such a case, all controls and stats are available, as shown above.

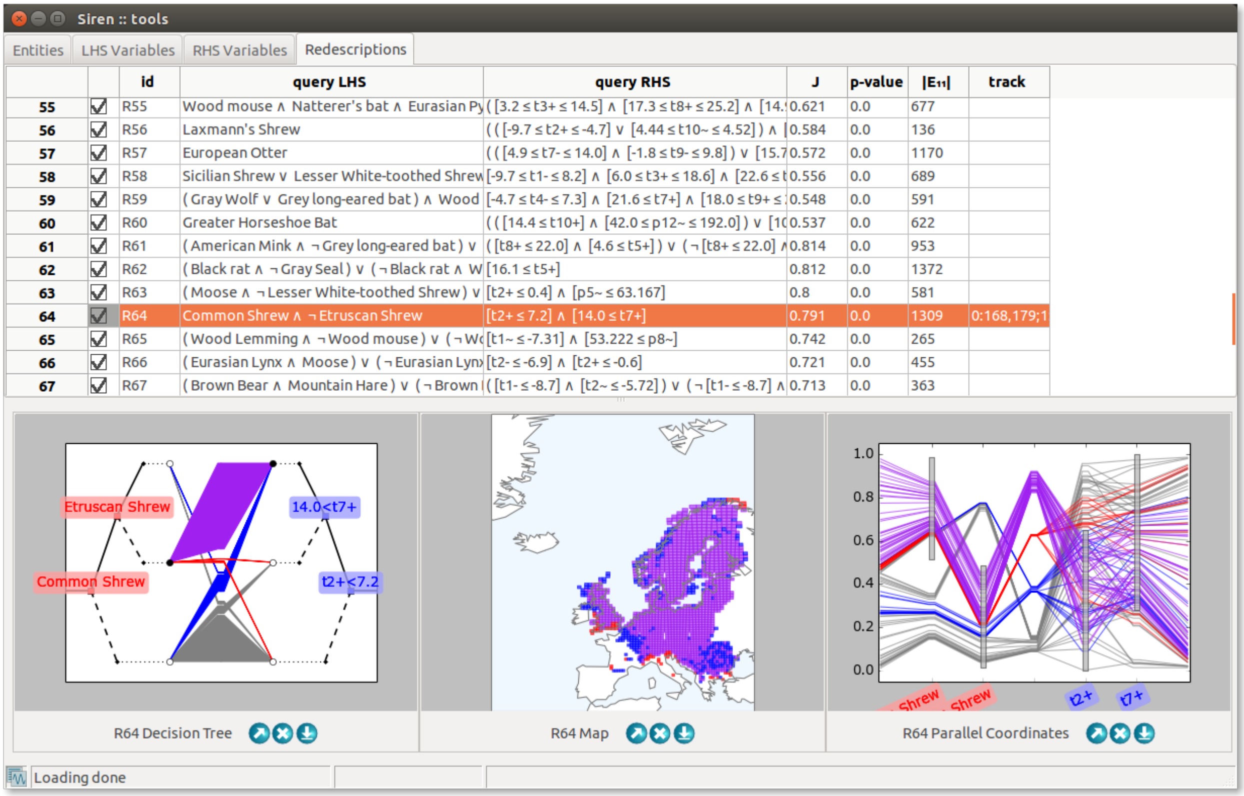
An alternative is to plot the visualization to a grid inside a dedicated tab of the main window. In that case, to accomodate more visualizations, controls are not shown, but the queries and stats are still visible.

An entry of the `Edit` menu allows to choose whether visualizations should be plotted by default into new separate windows or inside the visualization tab.

The plots can be popped out of the tab into separate windows, and vice-versa using the arrow buttons. The cross allows to close the view. Rows and columns can be added and dropped from the visualization grid by clicking on the green bars to the bottom and to the right of the grid, and the red bars to the top and to the left of the grid respectively. On deletion of a row or column of the grid, all views it contains are closed. The grid cell where the next visualization will be plotted is highlighted, a free cell can be chosen by simply clicking on it. If no free cell is available, the next plot will replace the oldest views in the grid, effectively deleting it.



Finally, clicking on the colored button on the bottom-left corner of the main window allows to split the main window horizontally into two frames, so as to show the grid of visualization beneath a list, for instance the list of redescriptions. To save further space the stats and queries are not shown in that case. Clicking anew on the colored button restores the main window to a single frame.



		id	query LHS	query RHS	J	p-value	$\lvert E_{11}\rvert$	track
55	☑	R55	Wood mouse ∧ Natterer's bat ∧ Eurasian Py	([3.2 ≤ t3+ ≤ 14.5] ∧ [17.3 ≤ t8+ ≤ 25.2] ∧ [14.	0.621	0.0	677	
56	☑	R56	Laxmann's Shrew	(([-9.7 ≤ t2+ ≤ -4.7] ∨ [4.44 ≤ t10~ ≤ 4.52]) ∧ [	0.584	0.0	136	
57	☑	R57	European Otter	(([4.9 ≤ t7- ≤ 14.0] ∧ [-1.8 ≤ t9- ≤ 9.8]) ∨ [15.7	0.572	0.0	1170	
58	☑	R58	Sicilian Shrew ∨ Lesser White-toothed Shrew	[-9.7 ≤ t1- ≤ 8.2] ∧ [6.0 ≤ t3+ ≤ 18.6] ∧ [22.6 ≤ t	0.556	0.0	689	
59	☑	R59	(Gray Wolf ∨ Grey long-eared bat) ∧ Wood	[-4.7 ≤ t4- ≤ 7.3] ∧ [21.6 ≤ t7+] ∧ [18.0 ≤ t9+ ≤	0.548	0.0	591	
60	☑	R60	Greater Horseshoe Bat	(([14.4 ≤ t10+] ∧ [42.0 ≤ p12~ ≤ 192.0]) ∨ [10	0.537	0.0	622	
61	☑	R61	(American Mink ∧ ¬ Grey long-eared bat) ∨	([t8+ ≤ 22.0] ∧ [4.6 ≤ t5+]) ∨ (¬ [t8+ ≤ 22.0] ∧	0.814	0.0	953	
62	☑	R62	(Black rat ∧ ¬ Gray Seal) ∨ (¬ Black rat ∧ W	[16.1 ≤ t5+]	0.812	0.0	1372	
63	☑	R63	(Moose ∧ ¬ Lesser White-toothed Shrew) ∨	[t2+ ≤ 0.4] ∧ [p5~ ≤ 63.167]	0.8	0.0	581	
64	☑	R64	Common Shrew ∧ ¬ Etruscan Shrew	[t2+ ≤ 7.2] ∧ [14.0 ≤ t7+]	0.791	0.0	1309	0:168,179;1
65	☑	R65	(Wood Lemming ∧ ¬ Wood mouse) ∨ (¬ Wo	[t1~ ≤ -7.31] ∧ [53.222 ≤ p8~]	0.742	0.0	265	
66	☑	R66	(Eurasian Lynx ∧ Moose) ∨ (¬ Eurasian Lyn	[t2- ≤ -6.9] ∧ [t2+ ≤ -0.6]	0.721	0.0	455	
67	☑	R67	(Brown Bear ∧ Mountain Hare) ∨ (¬ Brown	([t1- ≤ -8.7] ∧ [t2~ ≤ -5.72]) ∨ (¬ [t1- ≤ -8.7] ∧	0.713	0.0	363	

Loading done

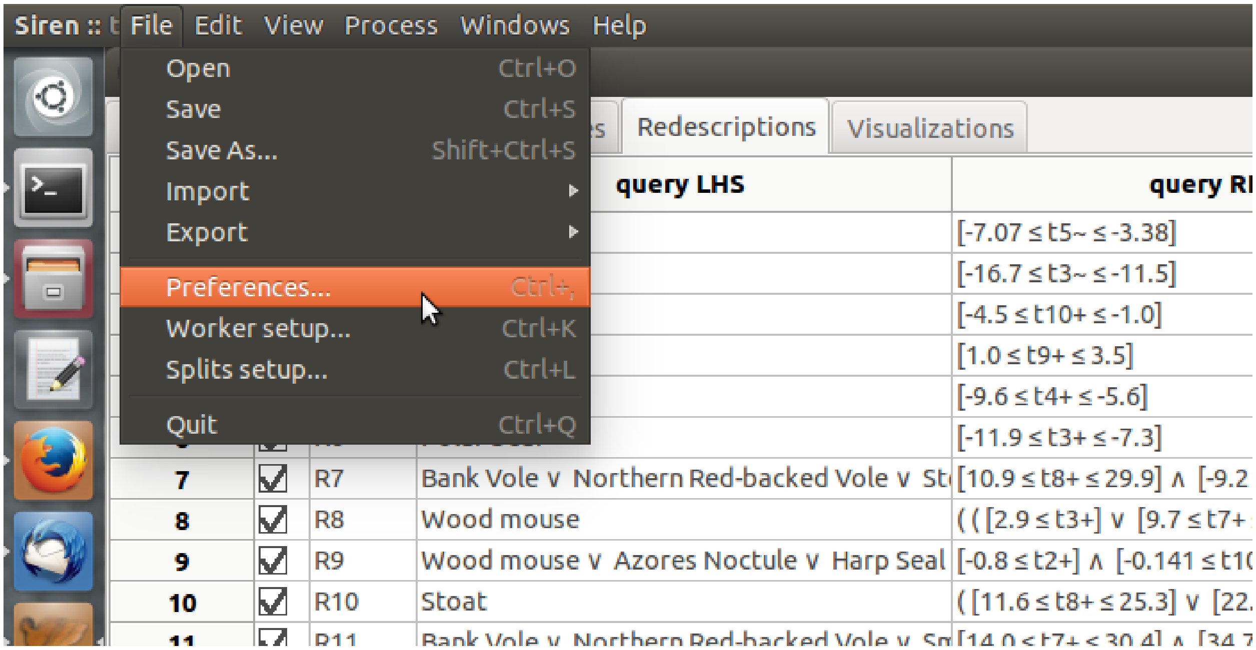
2.2 Menus

Note: The main menu of *Siren* is at the top of the *Tools window*.

Here is a summary of functionalities available throught the menu.

2.2.1 File

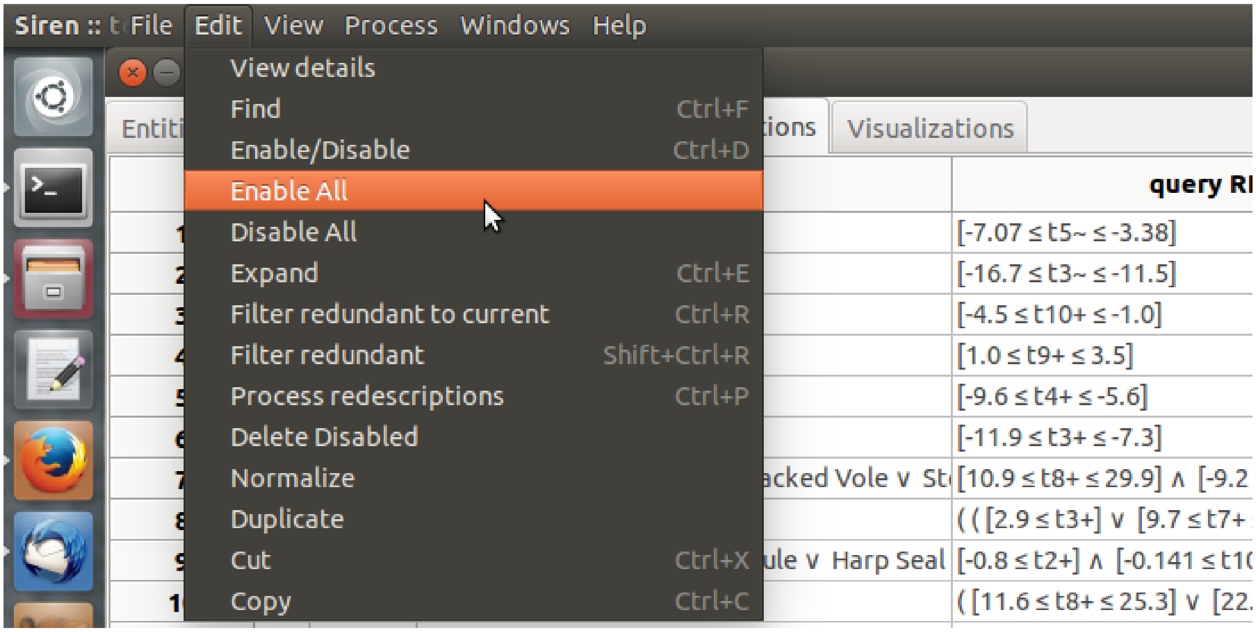
The `File` submenu provides import, export, opening and saving functionalities and setting the preferences.

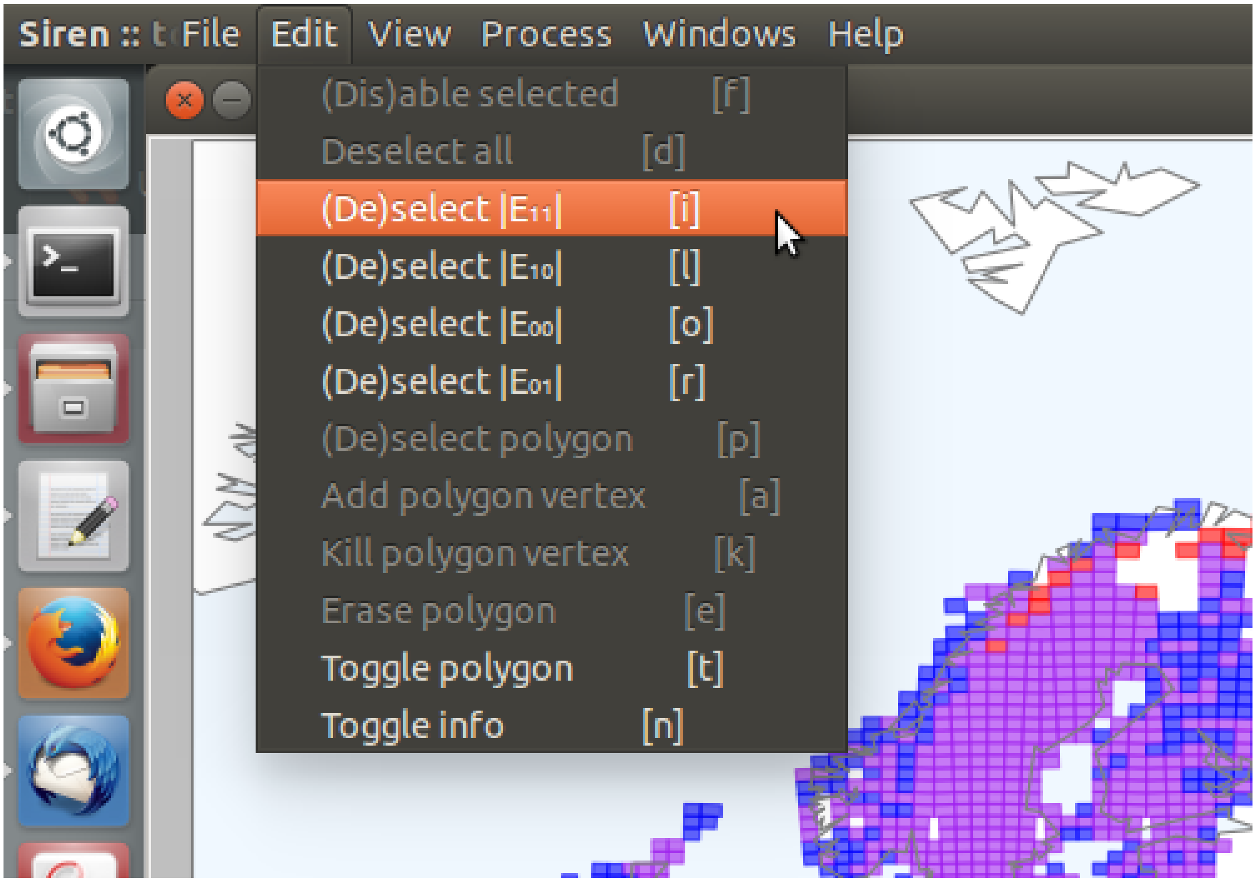


2.2.2 Edit

The content of the `Edit` submenu depends on the tab or view currently active.

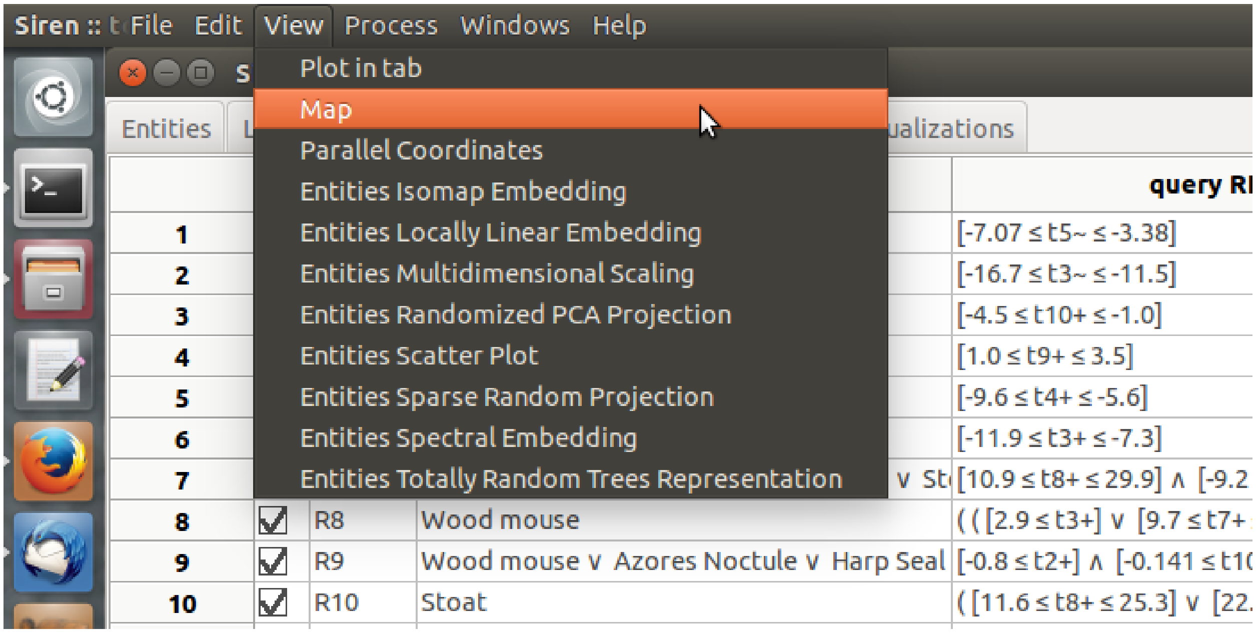
If the tab contains redescriptions, it will also allow to filter the redescription and to copy, cut and paste them.





2.2.3 Views

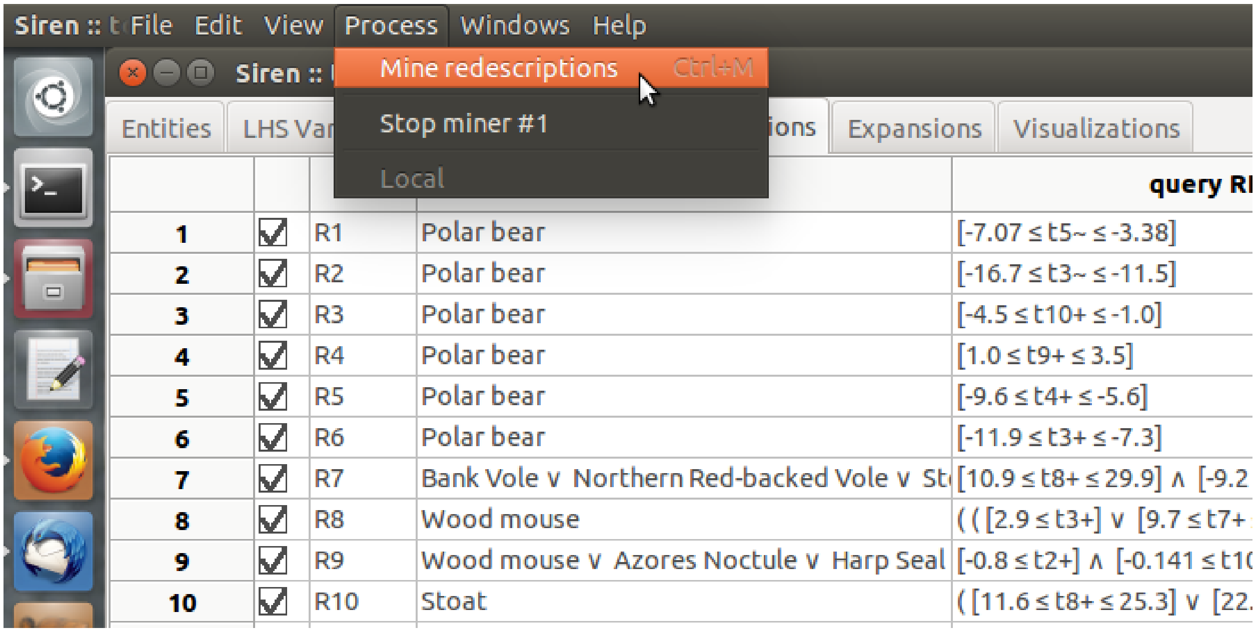
If the tab contains redescriptions or variables the `View` menu will allow to open a *view window* to visualize the selected item. Possible visualizations for an item might differ. For instance, if a redescription cannot be represented as a tree, the tree diagram entry of the menu will be disabled, the `normalize` function should be applied first.



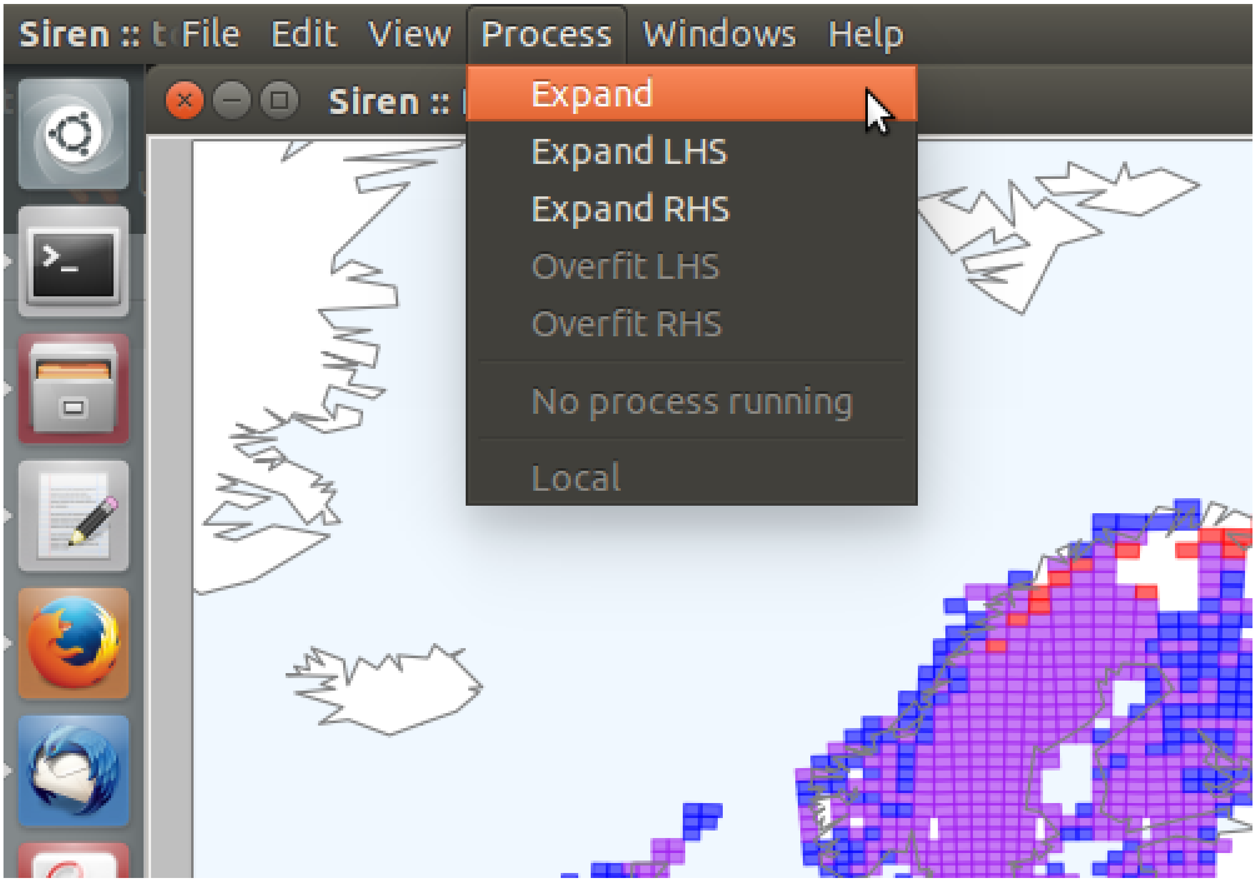
				query RI
1				[-7.07 ≤ t5~ ≤ -3.38]
2				[-16.7 ≤ t3~ ≤ -11.5]
3				[-4.5 ≤ t10+ ≤ -1.0]
4				[1.0 ≤ t9+ ≤ 3.5]
5				[-9.6 ≤ t4+ ≤ -5.6]
6				[-11.9 ≤ t3+ ≤ -7.3]
7			∨ St	[10.9 ≤ t8+ ≤ 29.9] ∧ [-9.2
8	☑	R8	Wood mouse	(([2.9 ≤ t3+] ∨ [9.7 ≤ t7+
9	☑	R9	Wood mouse ∨ Azores Noctule ∨ Harp Seal	[-0.8 ≤ t2+] ∧ [-0.141 ≤ t1
10	☑	R10	Stoat	([11.6 ≤ t8+ ≤ 25.3] ∨ [22.

2.2.4 Process

The `Process` menu allows to start mining redescriptions and contains a list of running task and allows to interrupt any of them.

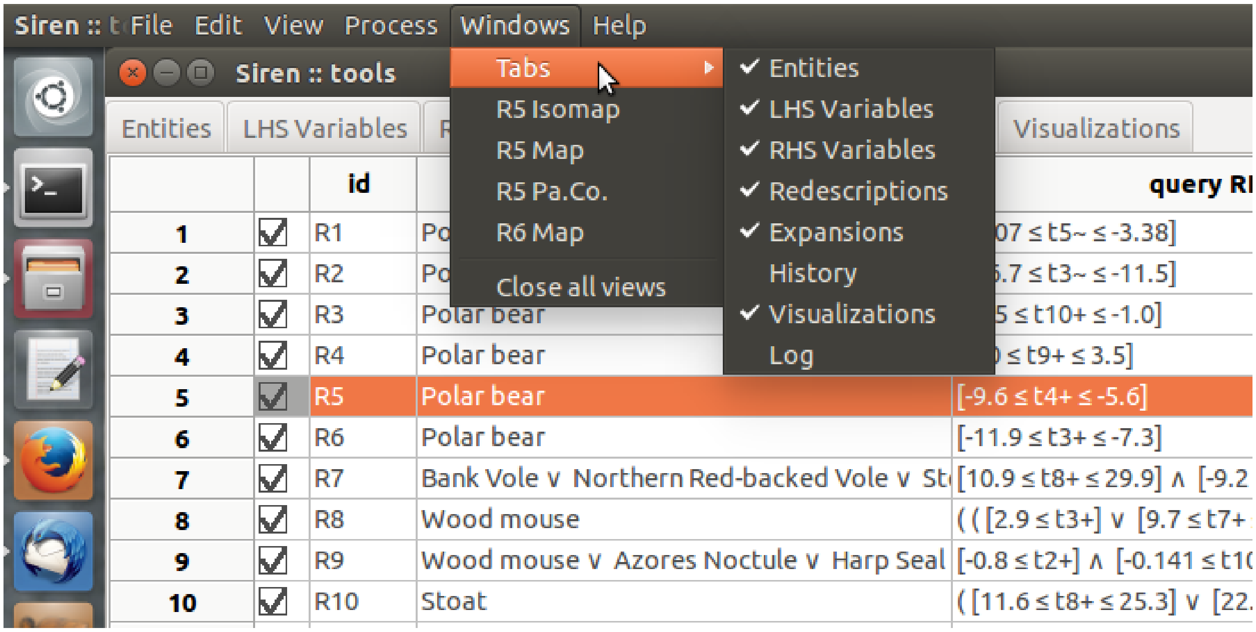


				query RI
1	☑	R1	Polar bear	[-7.07 ≤ t5~ ≤ -3.38]
2	☑	R2	Polar bear	[-16.7 ≤ t3~ ≤ -11.5]
3	☑	R3	Polar bear	[-4.5 ≤ t10+ ≤ -1.0]
4	☑	R4	Polar bear	[1.0 ≤ t9+ ≤ 3.5]
5	☑	R5	Polar bear	[-9.6 ≤ t4+ ≤ -5.6]
6	☑	R6	Polar bear	[-11.9 ≤ t3+ ≤ -7.3]
7	☑	R7	Bank Vole ∨ Northern Red-backed Vole ∨ St	[10.9 ≤ t8+ ≤ 29.9] ∧ [-9.2
8	☑	R8	Wood mouse	(([2.9 ≤ t3+] ∨ [9.7 ≤ t7+
9	☑	R9	Wood mouse ∨ Azores Noctule ∨ Harp Seal	[-0.8 ≤ t2+] ∧ [-0.141 ≤ t1
10	☑	R10	Stoat	([11.6 ≤ t8+ ≤ 25.3] ∨ [22.



2.2.5 Windows

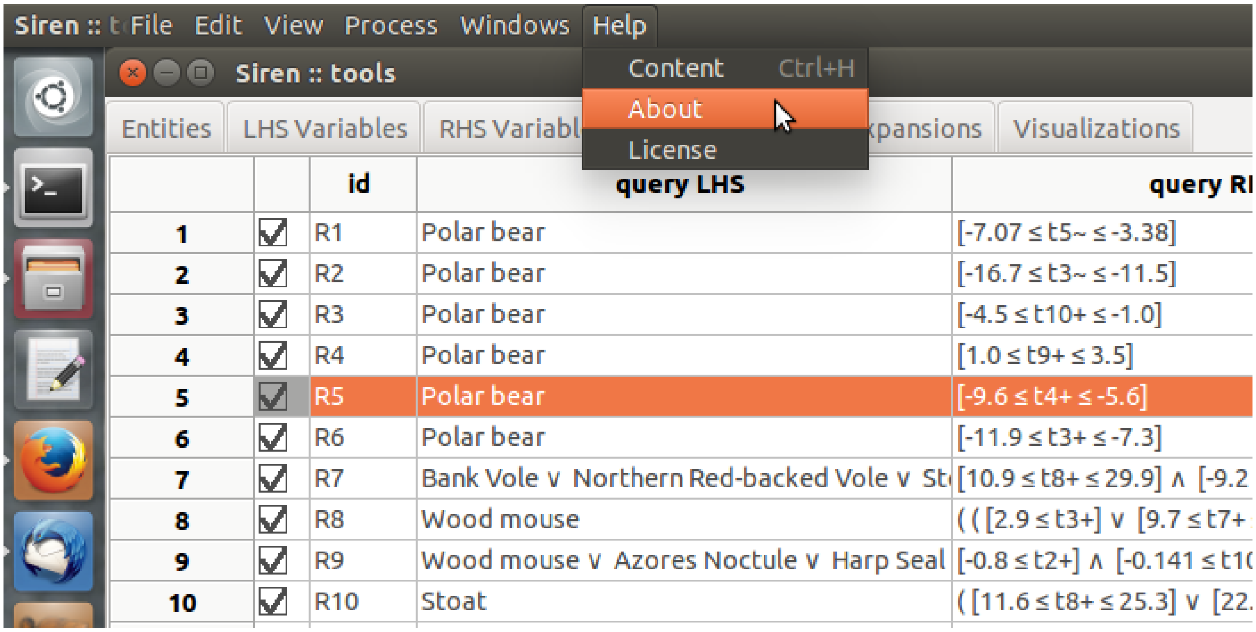
The `Windows` menu contains a list of tabs and allows to show or hide any of them. It also contains a list of Views currently opened sorted by redescription, allowing to access any of them and close all at once.



		id		query RI
1	☑	R1	Po	07 ≤ t5~ ≤ -3.38]
2	☑	R2	Po	.7 ≤ t3~ ≤ -11.5]
3	☑	R3	Polar bear	5 ≤ t10+ ≤ -1.0]
4	☑	R4	Polar bear	) ≤ t9+ ≤ 3.5]
5	☑	R5	Polar bear	[-9.6 ≤ t4+ ≤ -5.6]
6	☑	R6	Polar bear	[-11.9 ≤ t3+ ≤ -7.3]
7	☑	R7	Bank Vole ∨ Northern Red-backed Vole ∨ St	[10.9 ≤ t8+ ≤ 29.9] ∧ [-9.2
8	☑	R8	Wood mouse	(([2.9 ≤ t3+] ∨ [9.7 ≤ t7+
9	☑	R9	Wood mouse ∨ Azores Noctule ∨ Harp Seal	[-0.8 ≤ t2+] ∧ [-0.141 ≤ t1
10	☑	R10	Stoat	([11.6 ≤ t8+ ≤ 25.3] ∨ [22.

2.2.6 Help

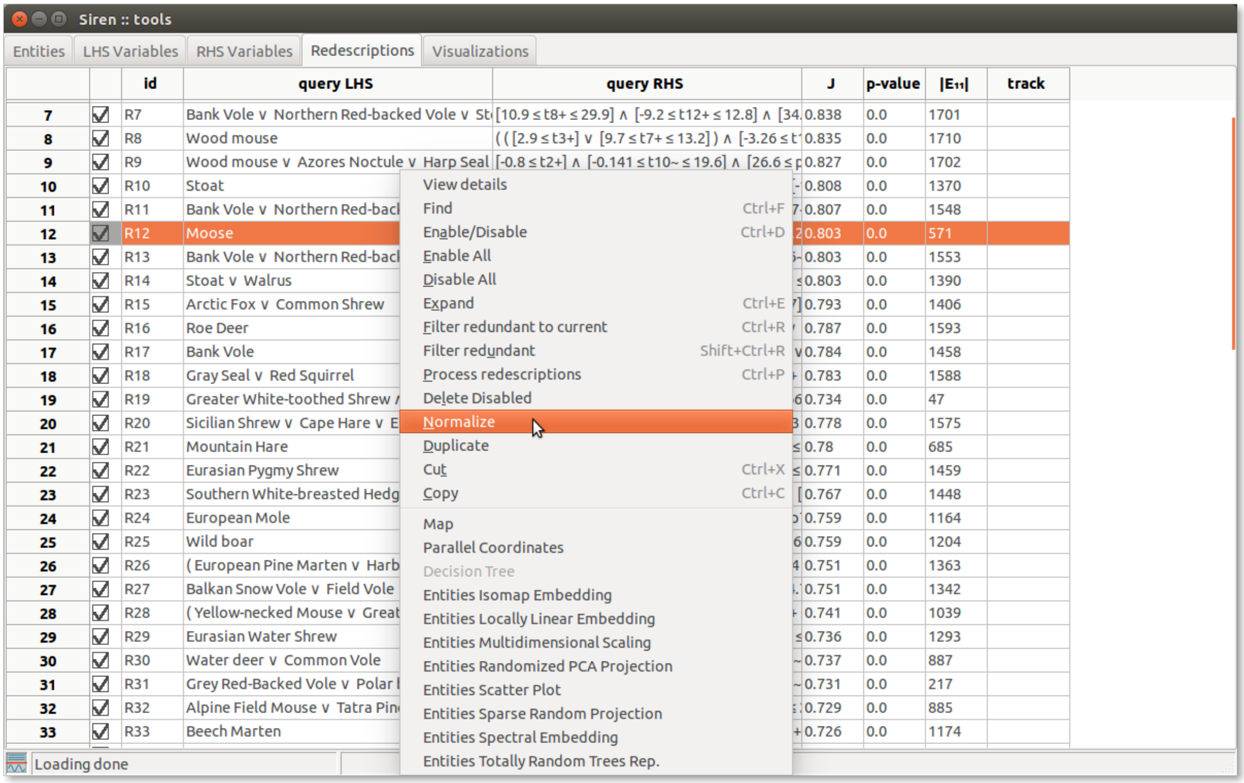
The `Help` menu provides access to this help and to more information about *Siren* and licensing.



		id	query LHS	query R
1	☑	R1	Polar bear	[-7.07 ≤ t5~ ≤ -3.38]
2	☑	R2	Polar bear	[-16.7 ≤ t3~ ≤ -11.5]
3	☑	R3	Polar bear	[-4.5 ≤ t10+ ≤ -1.0]
4	☑	R4	Polar bear	[1.0 ≤ t9+ ≤ 3.5]
5	☑	R5	Polar bear	[-9.6 ≤ t4+ ≤ -5.6]
6	☑	R6	Polar bear	[-11.9 ≤ t3+ ≤ -7.3]
7	☑	R7	Bank Vole ∨ Northern Red-backed Vole ∨ St	[10.9 ≤ t8+ ≤ 29.9] ∧ [-9.2
8	☑	R8	Wood mouse	(([2.9 ≤ t3+] ∨ [9.7 ≤ t7+
9	☑	R9	Wood mouse ∨ Azores Noctule ∨ Harp Seal	[-0.8 ≤ t2+] ∧ [-0.141 ≤ t1
10	☑	R10	Stoat	([11.6 ≤ t8+ ≤ 25.3] ∨ [22.

2.2.7 Contextual menu

Many `Edit` and `Views` functionalities can also be accessed via the contextual menu upon right-click on a redescription.



		id	query LHS	query RHS	J	p-value	$\lvert E_{11}\rvert$	track
7	☑	R7	Bank Vole ∨ Northern Red-backed Vole ∨ St	[10.9 ≤ t8+ ≤ 29.9] ∧ [-9.2 ≤ t12+ ≤ 12.8] ∧ [34	0.838	0.0	1701	
8	☑	R8	Wood mouse	(([2.9 ≤ t3+] ∨ [9.7 ≤ t7+ ≤ 13.2]) ∧ [-3.26 ≤ t	0.835	0.0	1710	
9	☑	R9	Wood mouse ∨ Azores Noctule ∨ Harp Seal	[-0.8 ≤ t2+] ∧ [-0.141 ≤ t10~ ≤ 19.6] ∧ [26.6 ≤ p	0.827	0.0	1702	
10	☑	R10	Stoat		0.808	0.0	1370	
11	☑	R11	Bank Vole ∨ Northern Red-back		0.807	0.0	1548	
12	☑	R12	Moose		0.803	0.0	571	
13	☑	R13	Bank Vole ∨ Northern Red-back		0.803	0.0	1553	
14	☑	R14	Stoat ∨ Walrus		0.803	0.0	1390	
15	☑	R15	Arctic Fox ∨ Common Shrew		0.793	0.0	1406	
16	☑	R16	Roe Deer		0.787	0.0	1593	
17	☑	R17	Bank Vole		0.784	0.0	1458	
18	☑	R18	Gray Seal ∨ Red Squirrel		0.783	0.0	1588	
19	☑	R19	Greater White-toothed Shrew		0.734	0.0	47	
20	☑	R20	Sicilian Shrew ∨ Cape Hare ∨ E		0.778	0.0	1575	
21	☑	R21	Mountain Hare		0.78	0.0	685	
22	☑	R22	Eurasian Pygmy Shrew		0.771	0.0	1459	
23	☑	R23	Southern White-breasted Hedg		0.767	0.0	1448	
24	☑	R24	European Mole		0.759	0.0	1164	
25	☑	R25	Wild boar		0.759	0.0	1204	
26	☑	R26	(European Pine Marten ∨ Harb		0.751	0.0	1363	
27	☑	R27	Balkan Snow Vole ∨ Field Vole		0.751	0.0	1342	
28	☑	R28	(Yellow-necked Mouse ∨ Great		0.741	0.0	1039	
29	☑	R29	Eurasian Water Shrew		0.736	0.0	1293	
30	☑	R30	Water deer ∨ Common Vole		0.737	0.0	887	
31	☑	R31	Grey Red-Backed Vole ∨ Polar		0.731	0.0	217	
32	☑	R32	Alpine Field Mouse ∨ Tatra Pin		0.729	0.0	885	
33	☑	R33	Beech Marten		0.726	0.0	1174	

Loading done

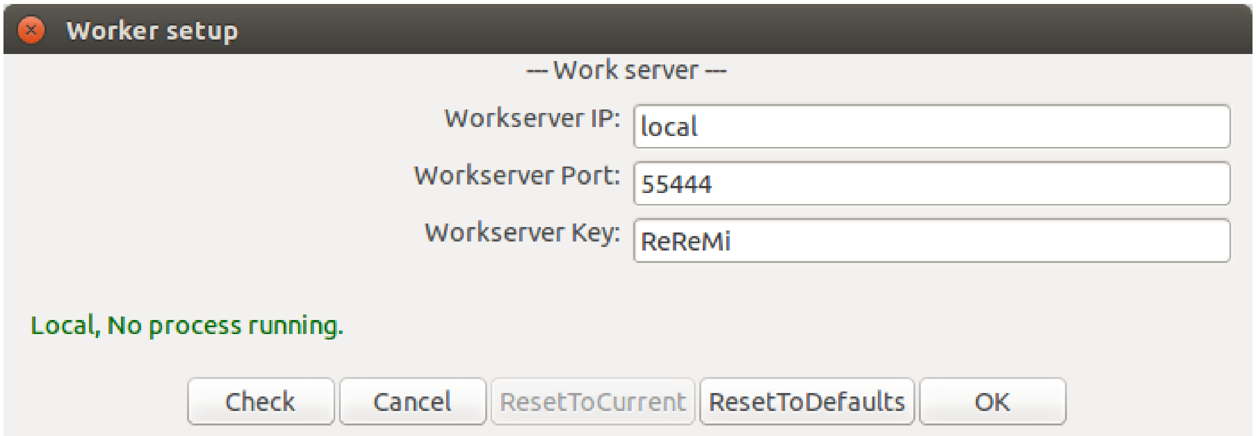
2.3 More

Note: *Siren* provide means to send computations to an external server, thereby helping to maintain the responsiveness of the interface

In addition, to help study how well the redescriptions mined using a particular set of parameters generalize to unseen entities, *Siren* allows to easily partition the data, run the algorithm on a selected subset of the data and compare accuracy and support of the results.

2.3.1 Mining server

To pass mining and projection computation on to an external server, the settings for communicating with that server can be set in the `Worker setup...` in the `File` menu. This allows to set the IP address (set to local to disable the client-server mode), port and authentication key, and to test the connection and check whether some processes are already being run on the server.



Workserver IP: local

Workserver Port: 55444

Workserver Key: ReReMi

Local, No process running.

Of course, the mining server should be setup with the relevant settings using the provided `server_siren.py` script.

Running the following, for instance, initializes a server on the feedback-loop listening on port 55444 with authentication key `atk` and having 4 workers (i.e. handling at most 4 different delegated tasks simultaneously).

```
python server_siren.py 127.0.0.1 --portnum=55444 --authkey=atk --max_k=4
```

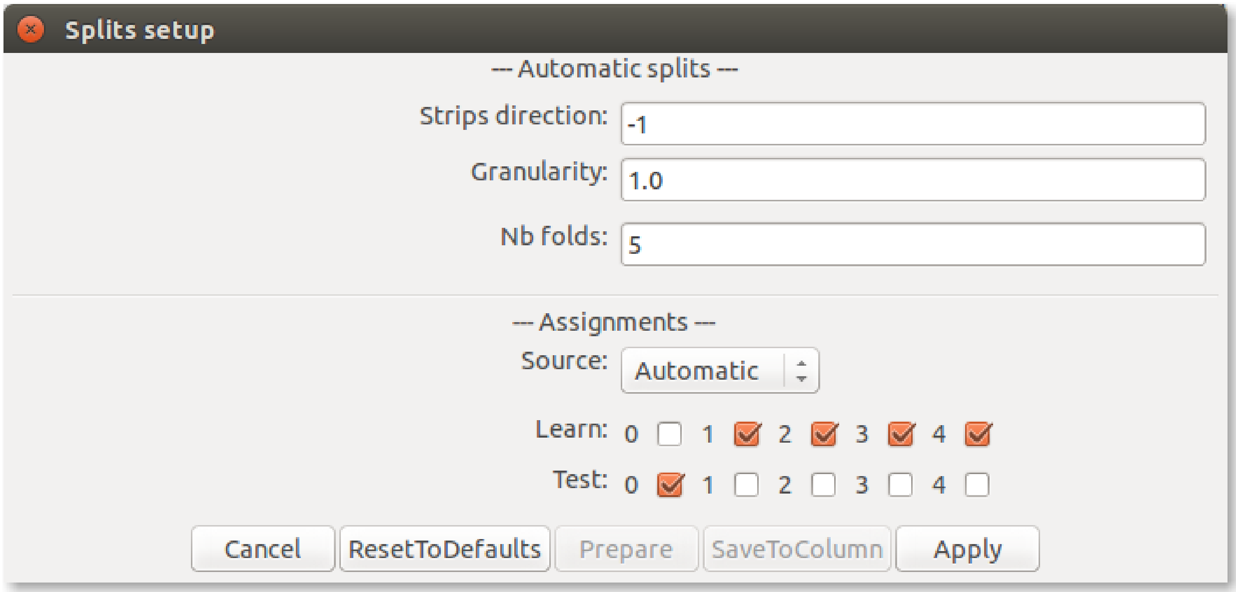
2.3.2 k-fold mining

Finally, *Siren* also allows to partition the data to study how redescription characteristics vary on different data subsets.

The settings for doing this can be accessed through the `Splits setup...` in the `File` menu.

The upper half of the dialogue box allows to choose parameters to generate a data partition automatically, by selecting a column or coordinate to whose values will direct the partition. A value greater than zero identifies a data column by its position, while a value lower than zero indentifies a coordinate, a value of zero means that the data will be partitioned randomly. That is, 1 corresponds to the first column on the left-hand side, while 12 corresponds to second column on the right-hand side assuming the left-hand side contains 10 columns, and -1 corresponds to the first coordinates dimension, i.e. the longitudes. The granularity allows to tune the width of the bands, while the number of folds defines the number of partitions returned.

The lower half of the dialogue box allows to choose the source, either desactivating the splits, using the automatic settings described above to generate partitions or extracting it from a data column (a categorical variable, where each category will generate a partition). Once this is set, the partitions can be generated. Next, the resulting partitions can be assigned to the learn (L) or test (T) folds. A partition can be stored in the data by encoding it as an additional column, so that it can be loaded again later on.



Strips direction: -1

Granularity: 1.0

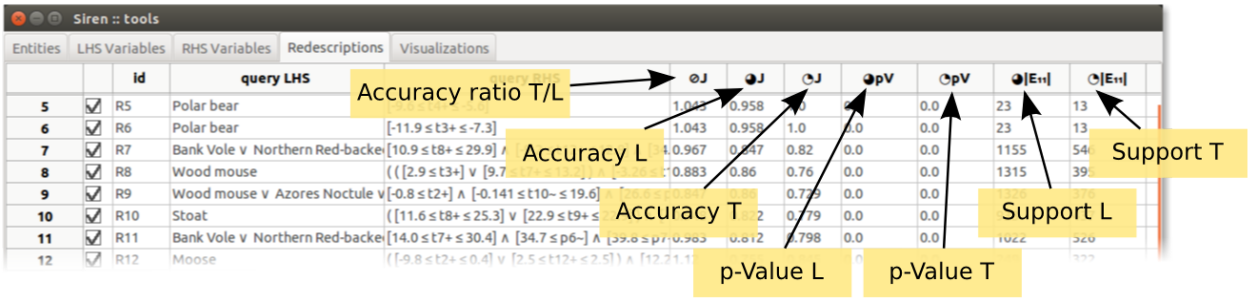
Nb folds: 5

Source: Automatic

Learn: 0 ☐ 1 ☑ 2 ☑ 3 ☑ 4 ☑

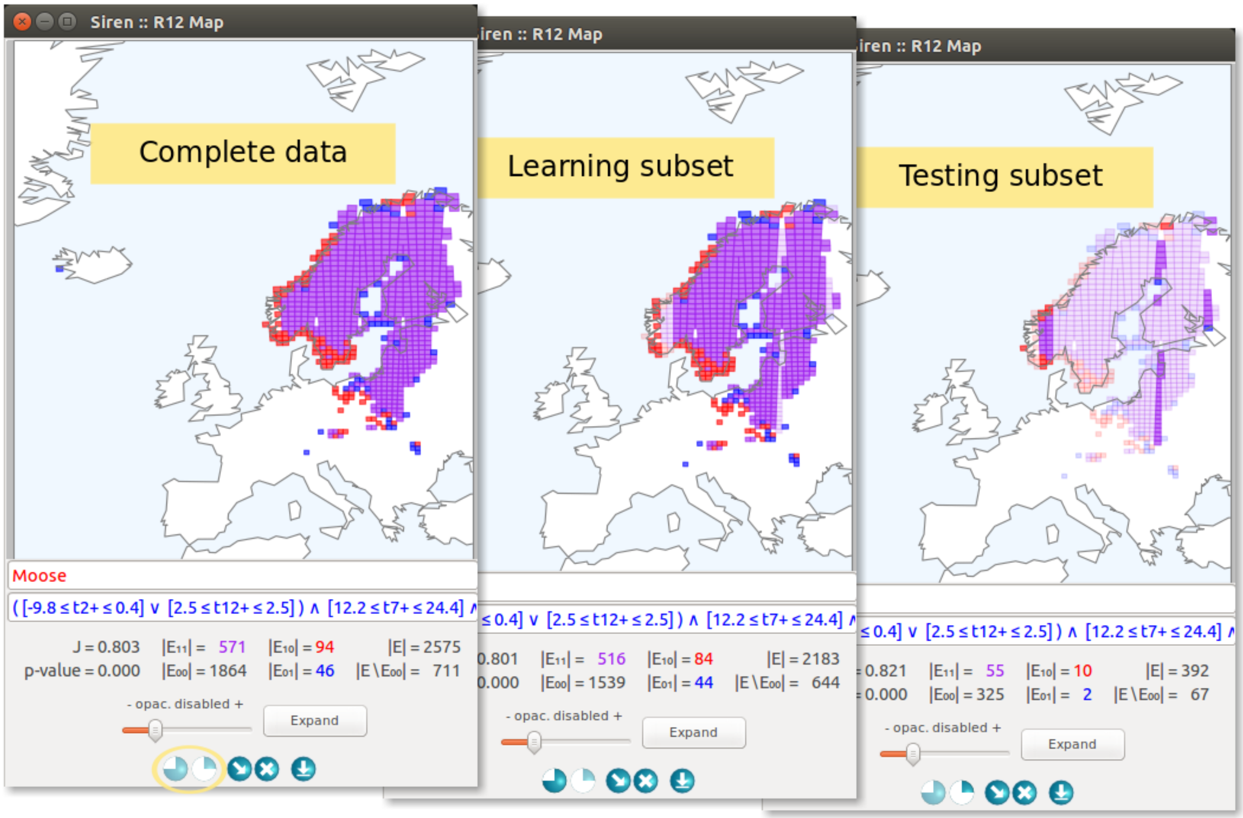
Test: 0 ☑ 1 ☐ 2 ☐ 3 ☐ 4 ☐

After the splits setup as been applied, new columns will appear in the redescription lists, showing the statistics (accuracy, p-Values, support) on the learn and test folds.



		id	query LHS	query RHS	⊘J	◕J	◔J	◕pV	◔pV	◕$\lvert E_{11}\rvert$	◔$\lvert E_{11}\rvert$
5	☑	R5	Polar bear	[-9.6 ≤ t4+ ≤ -5.6]	1.043	0.958	[illegible]	[illegible]	0.0	23	13
6	☑	R6	Polar bear	[-11.9 ≤ t3+ ≤ -7.3]	1.043	0.958	1.0	0.0	0.0	23	13
7	☑	R7	Bank Vole ∨ Northern Red-backe	[10.9 ≤ t8+ ≤ 29.9] ∧ [illegible]	0.967	0.847	0.82	0.0	0.0	1155	54
8	☑	R8	Wood mouse	(([2.9 ≤ t3+] ∨ [9.7 ≤ t7+ ≤ 13.2]) ∧ [-3.26 ≤ t	0.883	0.86	0.76	0.0	0.0	1315	39
9	☑	R9	Wood mouse ∨ Azores Noctule ∨	[-0.8 ≤ t2+] ∧ [-0.141 ≤ t10~ ≤ 19.6] ∧ [26.6 ≤ p	0.847	0.86	0.729	0.0	0.0	1326	376
10	☑	R10	Stoat	([11.6 ≤ t8+ ≤ 25.3] ∨ [22.9 ≤ t9+ ≤	[illegible]	[illegible]	0.779	0.0	0.0	[illegible]	[illegible]
11	☑	R11	Bank Vole ∨ Northern Red-backe	[14.0 ≤ t7+ ≤ 30.4] ∧ [34.7 ≤ p6~] ∧ [39.8 ≤ p7	0.983	0.812	0.798	0.0	0.0	1022	526
12	☑	R12	Moose	([-9.8 ≤ t2+ ≤ 0.4] ∨ [2.5 ≤ t12+ ≤ 2.5]) ∧ [12.2	1.1	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	322

In views, a pair of buttons will appear allowing to focus on the complete data or on either split.



Moose

([-9.8 ≤ t2+ ≤ 0.4] ∨ [2.5 ≤ t12+ ≤ 2.5]) ∧ [12.2 ≤ t7+ ≤ 24.4] ∧

J = 0.803 |E_{11}| = 571 |E_{10}| = 94 |E| = 2575
p-value = 0.000 |E_{00}| = 1864 |E_{01}| = 46 |E \ E_{00}| = 711

≤ 0.4] ∨ [2.5 ≤ t12+ ≤ 2.5]) ∧ [12.2 ≤ t7+ ≤ 24.4] ∧

0.801 |E_{11}| = 516 |E_{10}| = 84 |E| = 2183
0.000 |E_{00}| = 1539 |E_{01}| = 44 |E \ E_{00}| = 644

≤ 0.4] ∨ [2.5 ≤ t12+ ≤ 2.5]) ∧ [12.2 ≤ t7+ ≤ 24.4] ∧

0.821 |E_{11}| = 55 |E_{10}| = 10 |E| = 392
0.000 |E_{00}| = 325 |E_{01}| = 2 |E \ E_{00}| = 67

CHAPTER THREE

FORMATS

3.1 Data formats

Note: For redescription mining, one considers entities discribed by variables divided into two sets, hereafter arbitrarily called left-hand side and right-hand side. This can be seen as a pair of data matrices, where entities are identified with rows and variables with columns. Both sets of variables describe the same entities, hence, the matrices have the same number of rows.

If you provide the same dataset for the left and right hand sides, this will be interpreted as a settings with a single datasets, where variables can appear on either side of redescriptions, but not both in the same redescription. Variables can be selectively disabled on either side, to prevent them from being used in either query.

In *Siren*, data include:

- **Variables**: The variables describing the entities are divided in two sets. They can be of three types:
    1. Boolean,
    2. categorical,
    3. or real-valued.

Obviously, this is required.

- **Entities names**: Optional additional information, providing names for the entities.
- **Variable names**: Optional additional information, providing names for the variables.
- **Coordinates**: Optional location information, i.e. geographic coordinates of the entities. This makes the data geospatial.

Data can be imported to *Siren* via the interface menu *File* → *Import* → *Import Data*. Below, we present the data formats supported by *Siren*.

Data can be imported into *Siren* as CSV files. The program expects a pair of files, one for either side in character-separated values, as can be imported and exported to and from spreadsheet programms, for instance.

There are two main formats,

- **Full**: standard table format, or
- **Sparse**: compact format for dataset with few non-zeros entries.

The two data files need not be in the same format.

If entities names and/or coordinates are provided, they will be used to match entities across the two sides. Otherwise, rows will be match in order and an error will occur if the two side do not contain the same number of rows.

3.1.1 Full format

The data is stored as a table with one column for each variable and one row each entity. The first row can contain the names of the variables. The entities names can be included as columns named *id*. Similarly the coordinates can be included as a pair of columns named *longitude* and *latitude*, respectively.

3.1.2 Sparse format

This format allows to store data that contains few non-zero entries more compactly, as in the Matlab sparse format (or like the edge list of a bipartite graph).

Each line contains an entry of the data as a triple (entity, variable, value). This way, the data is stored as in three columns and as many rows as there are entries. In this case the first line of the data file must contains *id*, *cid* and *value*, indicating the three columns containing the enities, variables and corresponding value, respectively. Coordinates can be provided in a similar way under the variable names *longitude* and *latitude*.

Variable names can be provided inline, that is, simply by using the name of the variable for each entry involving it. Alternatively, variable names can be specified separately with a special "-1" entity. Similarly, entity names can be provided inline or separatly with a special "-1" variable. For example, the following five lines

```
id; cid; value
Espoo; population; 260981
Helsinki; population; 614074
Tampere; population; 220609
Turku; population; 182281
```

are equivalent to the following:

```
id; cid; value
20; -1; Espoo
7; -1; Tampere
2; -1; Turku
13; -1; Helsinki
-1; 3; population
2; 3; 182281
7; 3; 220609
13; 3; 614074
20; 3; 260981
```

Finally, in case of fully Boolean data without coordinates, the value can be left out. Each pair of (entity, variable) appearing is considered as True, the rest as False.

For both full and sparse formats a mention of type can be append to the first row, in such case all variable will be parse to the given type. For instance, in the example above the first line would be turned to `id; cid; value; type=N` to ensure that all variables, including population are interpreted as numerical (N) variables. Respectively B and C can be used to ensure that all variables are Boolean and categorical, respectively.

This can be useful when handling a dataset of numerical variables where some contains only two distinct values and might otherwise be interpreted as Boolean variables. It can also be a handy way to turn a dataset to fully Boolean based on zero/non-zero values. However, be warned that this can cause some troubles…

3.2 Redescriptions formats

Note: The product of redescription mining is a list of redescriptions. A redescription consist of a pair of queries over the variables describing the entities, one query for each set. The two sets of variables are arbitrarily called left-hand side and right-hand side, and so are the corresponding queries.

3.2.1 Supports

The support of a query is the set of entities for which the query holds. Any given redescription partitions the entities into four sets (In the absence of missing entries):

- E_{10} is the set of rows for which only the left hand side query holds,
- E_{01} is the set of rows for which only the right hand side query holds,
- E_{11} is the set of rows for which both queries hold,
- and E_{00} is the set of rows for which neither of the queries hold.

Redescriptions can be imported to *Siren* via the interface menu *File* → *Import* → *Import Redescriptions*. More importantly, they can be exported via the interface menu *File* → *Export Redescriptions* and the contextual menu for a list of redescription. Below, we present the redescription formats supported by *Siren*.

3.2.2 Queries

A query is formed by combining literal using Boolean operators.

While *ReReMi* only generate linearly parsable query (see references for more details), *Siren* can actually evaluates arbitrary queries, as long as they are well formed following the informal grammar below. In particular, parenthesis should be used to separated conjunctive blocks and disjunctive block, alternating between operators. For example, while the later cannot be generated by *ReReMi*, $(a \wedge b) \vee \neg c$ and $(a \wedge b) \vee (c \wedge d)$ are both supported. $(a \wedge b) \wedge (c \wedge d)$ is not, because of incorrect alternance of operators between parenthesis blocks. It should simply be written as $a \wedge b \wedge c \wedge d$.

We consider three types of literals, defined over a Boolean, categorical or numerical variable respectively.

Below is an unformal grammar of *Siren*'s query language. The actual grammar can be found in the `redquery.ebnf` file in the `siren.reremi` source repertory.

Tip:

query = disjunction | conjunction | literal ;

conjunction = conj_item { (“&” | ” $\wedge$ ”) conj_item }+ ;

disjunction = disj_item { (“|” | ” $\vee$ ”) disj_item }+ ;

conj_item = literal | (“(” disjunction “)”) ;

disj_item = literal | (“(” conjunction “)”) ;

literal = categorical_literal | realvalued_literal | boolean_literal ;

categorical_literal = (“[”)? variable_name (” $=$ ” | ” $\neq$ ” | ” $\in$ ” | ” $\in$ ”) category (“]”)? ;

realvalued_literal = [neg] (“[”)? [variable_value lth] variable_name lth variable_value (“]”)? ;

realvalued_literal = [neg] (“[”)? variable_value lth variable_name (“]”)? ;

boolean_literal = [neg] (“[”)? variable_name (“]”)? ;

variable_name = STRING | ?/vd+/? ;

category = STRING | ?/d+/? ;

variable_value = ?/[+-]?d+([.])?d*([Ee][-+]d+)?/? ;

lth = “<” | ” $\leq$ ” ;

neg = “!” | ”¬” ;

Naturally, the type of literal and the type of variable should match, i.e., $[4.0 \leq Va \leq 8.32]$ is a valid numerical literal only if the corresponding variable Va is a numerical variable. Furthermore, the upper bound of a numerical variable should always be greater or equal to the lower bound and either of them should be specified.

3.2.3 Redescription statistics

The statistics of a redescription include:

- accuracy, as measured by Jaccard coefficient $|E_{11}|/(|E_{10}| + |E_{11}| + |E_{01}|)$,
- p-value,
- cardinality of the *support sets* E_{10}, E_{01}, E_{11}, E_{00} (sometimes also referred to as alpha, beta, gamma and delta, respectively).

3.2.4 Exporting Redescriptions

Redescriptions from the `Redescriptions` tab can be exported to a file, one redescription per line, with both queries and basic statistics tab separated. Three of formatting options are available, determined by the provided filename:

- **named**: Uses the names of the variables instead of variable ids in the queries. Activated if the filename matches the pattern `*[^a-zA-Z0-9]named[^a-zA-Z0-9]*`.
- **support**: Prints out the support sets. Activated if the filename matches the pattern `*[^a-zA-Z0-9]support[^a-zA-Z0-9]*`.
- **all** By default disabled redescriptions will not be printed when exporting redescriptions. If the filename matches the pattern `*[^a-zA-Z0-9]all[^a-zA-Z0-9]*`, disabled redescriptions will also be printed.
- **tex** Rather than tab separated format, if the filename as `.tex` extension, a tex file is produced that can be compiled to obtain a table of the redescriptions. Three table layouts are available, where the information for each redescription is listed respectively on one, two or three rows, if the filename matches the pattern `*[^a-zA-Z0-9][1-3].[a-z]*$`. Note that this format cannot be imported back.

Inside a siren package, the redescriptions are stored in tab separated format.

The fields included when exporting redescriptions and when displaying them in the interface can be set via the *File* → *Fields setup* menu entries.

3.2.5 Importing Redescriptions

Tab separated formats can be imported into *Siren*, *TeX* cannot.

CHAPTER

FOUR

PARAMETERS

Note: Siren allows to tune various parameters via an interface menu *Edit* → *Preferences*.

Non-default parameters can be stored in a preferences file, e.g. via *Export* → *Export preferences*.

Parameters are listed below by their label and identifier, as used respectively in the interface and in preference files, with a short explanation of the parameter, type, default value and indication of file where the parameter is defined.

The parameters that can be set through the preference interface can be mainly divided into two categories, **mining parameters**, that pertain to the mining algorithm, and **visualization parameters**, that pertain to the interactive visualizations, each split over several tabs.

4.1 Mining parameters

Mining parameters are specified here and can be set through the interface.

Quite many parameters are available for tuning, but in general only a few of them need to be adjusted. In particular, the parameters setting threshold values on the support of the queries, on the accuracy of the returned redescriptions, as well as on the type of queries, are the most important ones.

Mining: Method

Mining algorithm
`mining_algo`

- Algorithm to use for mining, greedy ReReMi or based on classification trees (restrictions apply).
- single option in {reremi, layeredtrees, splittrees, cartwheel, fim}
- default: reremi
- miner_confdef.xml

Mining: Queries

LHS max number of variables
`max_var_s0`

- Maximum number of variables in a left-hand side query, maximum tree depth in tree-based algorithms.
- open text

- default: 4
- miner_confdef.xml

RHS max number of variables
`max_var_s1`

- Maximum number of variables in a right-hand side query, maximum tree depth in tree-based algorithms.
- open text
- default: 4
- miner_confdef.xml

Mining: Candidate filtering

Min contribution
`min_itm_c`

- Minimum contribution, number of supporting entities impacted by the addition of a literal in ReReMi or minimum node size in tree-based algorithms. Values between 0 and 1 are interpreted as ratio of the total number of entities.
- open text
- default: 3.0
- miner_confdef.xml

Min support
`min_itm_in`

- Minimum number of supporting entities. Values between 0 and 1 are interpreted as ratio of the total number of entities.
- open text
- default: 0.05
- miner_confdef.xml

Min uncovered
`min_itm_out`

- Minimum number of uncovered entities. Values between 0 and 1 are interpreted as ratio of the total number of entities.
- open text
- default: 0.05
- miner_confdef.xml

Min improvement
`min_impr`

- Minimum improvement score for an acceptable extension.
- open text
- default: 0.0
- miner_confdef.xml

Max initial candidates

`max_inits`

- Maximum number of initial candidates explored, pairs for ReReMi, initial targets for trees. -1 leads to expand all.
- open text
- default: 100
- miner_confdef.xml

Initial candidates productivity
`inits_productivity`

- Productivity when creating an initial pair of literals from non-Boolean variables.
- single option in {low, medium, high}
- default: medium
- miner_confdef.xml

Excluded initial cand. pair types
`inits_types_exclude`

- Excluded variable types in initial pairs. B, C and N stand for Boolean, Categorical and Numerical variables, respectively. Note order is simpler type first, not by sides.
- multiple options in {BB, BC, BN, CC, CN, NN}
- default: {}
- miner_confdef.xml

Mining: Final filtering

Min support
`min_fin_in`

- Minimum number of supporting entities, applied for final filtering. Values between 0 and 1 are interpreted as ratio of the total number of entities.
- open text
- default: -1.0
- miner_confdef.xml

Min uncovered
`min_fin_out`

- Minimum number of uncovered entities, applied for final filtering. Values between 0 and 1 are interpreted as ratio of the total number of entities.
- open text
- default: -1.0
- miner_confdef.xml

Min accuracy
`min_fin_acc`

- Minimum accuracy, applied for final filtering.
- open text

- default: 0.01
- miner_confdef.xml

Max p-value
`max_fin_pval`

- Maximum p-value, applied for final filtering.
- open text
- default: 0.01
- miner_confdef.xml

Mining: Additional filtering

Max Area Overlap
`max_overlap_area`

- Two redescriptions whose area overlap is beyond this ratio are considered redundant.
- open text
- default: 0.5
- miner_confdef.xml

Max Rows Overlap
`max_overlap_rows`

- Two redescriptions whose row overlap is beyond this ratio are considered redundant.
- open text
- default: 0.5
- miner_confdef.xml

Mining adv.: Process

Nb processes
`nb_processes`

- Maximum number of mining processes to use in parallel.
- open text
- default: 1
- miner_confdef.xml

Log verbosity
`verbosity`

- Verbosity of the mining process output.
- range in [0, 10]
- default: 4
- miner_confdef.xml

Debug checks
`debug_checks`

- For debugging, double check counts.
- yes/no
- default: no
- miner_confdef.xml

Mining adv.: Other common parameters

Batch output
`batch_out`

- Maximum number of redescriptions returned from an expansion batch.
- open text
- default: 1
- miner_confdef.xml

Min total number of variables
`min_fin_var`

- Maximum number of variables per redescription, applied for final filtering.
- open text
- default: 2
- miner_confdef.xml

Weight Inside
`in_weight`

- Weight for entities inside selected area (only applies when such an area is specified).
- open text
- default: 10
- miner_confdef.xml

Weight Outside
`out_weight`

- Weight for entities outside selected area (only applies when such an area is specified).
- open text
- default: 1
- miner_confdef.xml

Var group compatibility
`var_compat`

- Fixes whether group ids should differ or match for variables to be considered compatible (gid=-1 is universally compatible).
- single option in {differ, match}
- default: differ
- miner_confdef.xml

p-value formula

`method_pval`

- Formula used for computing the p-value.
- single option in {marg, supp, over}
- default: marg
- miner_confdef.xml

Missing values belief
`parts_type`

- Type of Jaccard index, determines how to consider missing values when computing support statistics.
- single option in {rejective, optimistic, pessimistic, positive, negative, basic, exclu}
- default: rejective
- miner_confdef.xml

Mining adv.: Tree-based algorithms

Max rounds in CartWheel
`max_rounds`

- Maximum number of alternations in CartWheel algorithm.
- open text
- default: 10
- miner_confdef.xml

Split criterion
`split_criterion`

- Score used to evaluate splits during decision tree induction.
- single option in {gini, entropy}
- default: gini
- miner_confdef.xml

Mining adv.: Splittrees variant

Init depth
`splittrees_init_depth`

- Maximum tree depth for the split in the first round.
- open text
- default: 1
- miner_confdef.xml

Update target
`splittrees_update_target`

- Update the target between tree growing rounds.
- yes/no
- default: no

- miner_confdef.xml

Both sides change
splittrees_both_sides_change

- Require both sides to be different in successive tree growing rounds.
- yes/no
- default: no
- miner_confdef.xml

Dynamic side
splittrees_dyn_side

- Choose the side on which to grow the tree first depending on the source of the target (else always start with RHS).
- yes/no
- default: yes
- miner_confdef.xml

ReReMi: Queries

Single side OR
single_side_or

- Restrict use of disjunction to only one side.
- yes/no
- default: yes
- miner_confdef.xml

Multi categories
multi_cats

- Allow combining multiple categories of a categorical variable.
- yes/no
- default: no
- miner_confdef.xml

Conditional
add_condition

- Allow making conditional redescriptions, if suitable data is provided.
- yes/no
- default: no
- miner_confdef.xml

LHS Literals Bool
neg_query_s0_1

- Types of literals allowed in the left-hand side queries for Boolean literals.
- multiple options in {positive, negative}

- default: {positive}
- miner_confdef.xml

LHS Literals Cat
`neg_query_s0_2`

- Types of literals allowed in the left-hand side queries for categorical literals.
- multiple options in {positive, negative}
- default: {positive}
- miner_confdef.xml

LHS Literals Num
`neg_query_s0_3`

- Types of literals allowed in the left-hand side queries for numerical literals.
- multiple options in {positive, negative}
- default: {positive}
- miner_confdef.xml

LHS Operators
`ops_query_s0`

- Type of operators allowed in the left-hand side queries.
- multiple options in {conjunction, disjunction}
- default: {conjunction, disjunction}
- miner_confdef.xml

RHS Literals Bool
`neg_query_s1_1`

- Types of literals allowed in the right-hand side queries for Boolean literals.
- multiple options in {positive, negative}
- default: {positive}
- miner_confdef.xml

RHS Literals Cat
`neg_query_s1_2`

- Types of literals allowed in the right-hand side queries for categorical literals.
- multiple options in {positive, negative}
- default: {positive}
- miner_confdef.xml

RHS Literals Num
`neg_query_s1_3`

- Types of literals allowed in the right-hand side queries for numerical literals.
- multiple options in {positive, negative}
- default: {positive}
- miner_confdef.xml

RHS Operators
`ops_query_s1`

- Type of operators allowed in the right-hand side queries.
- multiple options in {conjunction, disjunction}
- default: {conjunction, disjunction}
- miner_confdef.xml

ReReMi: Initial pairs

Init min score
`init_minscore`

- Minimum score for an initial candidate to be considered.
- open text
- default: 0.01
- miner_confdef.xml

Pairs-expand balance
`pe_balance`

- Balance between generating initial pairs and expanding them (0 means generating all pairs first).
- range in [0, 10]
- default: 4
- miner_confdef.xml

Expansion order
`pair_sel`

- Order in which the initial pairs are processed, either overall next best score or alternating between side considering next best variables in turn.
- single option in {overall, alternate}
- default: alternate
- miner_confdef.xml

ReReMi exp.: Candidate expansion

Accuracy improvement coefficient
`score.impacc`

- Coefficient of the accuracy improvement for scoring extension candidates.
- open text
- default: 1.0
- miner_confdef.xml

Relative Accurracy improvement coefficient
`score.rel_impacc`

- Coefficient of the relative accuracy improvement for scoring extension candidates.

- open text
- default: 0.0
- miner_confdef.xml

Redescription p-value coefficient
`score.pval_red`

- Coefficient of the redescription p-value for scoring extension candidates. Negative values are interpreted as a threshold weighted by p-value factor.
- open text
- default: 0.01
- miner_confdef.xml

Query p-value coefficient
`score.pval_query`

- Coefficient of the query p-value for scoring extension candidates. Negative values are interpreted as a threshold weighted by p-value factor.
- open text
- default: 0.01
- miner_confdef.xml

p-value factor
`score.pval_fact`

- Weight for p-value in case of using thresholds for scoring extension candidates.
- open text
- default: 1.0
- miner_confdef.xml

Batch capacity
`batch_cap`

- Maximum number of candidates expanded at each step of the search.
- open text
- default: 4
- miner_confdef.xml

Amnesic
`amnesic`

- Not amnesic means the algorithm remembers previously generated redescriptions and avoid generating them again.
- yes/no
- default: no
- miner_confdef.xml

ReReMi exp.: Numerical attributes discretization

Max number segments
max_seg

- Maximum number of segments, i.e. distinct values for a numerical variable.
- open text
- default: 20
- miner_confdef.xml

Lower tail aggregate
lower_tail_agg

- Threshold for the lower tail, when aggregating keeping only extreme values.
- open text
- default: 0.0
- miner_confdef.xml

Upper tail aggregate
upper_tail_agg

- Threshold for the upper tail, when aggregating keeping only extreme values.
- open text
- default: 0.0
- miner_confdef.xml

Max aggregation
max_agg

- Maximum number of buckets aggregated together.
- open text
- default: 15
- miner_confdef.xml

Max product buckets
max_prodbuckets

- Maximum product of the number of buckets on either side.
- open text
- default: 5000
- miner_confdef.xml

4.2 Visualization parameters

Visualization parameters are specified here and can be set through the interface. For instance, they allows to modify the appearance of the maps, choose the colors of the different support subsets, etc.

Views: Vizualizations

Intab grid rows
`intab_nbr`

- Number of rows of the grid for visualizations in the tab.
- open text
- default: 1
- views_confdef.xml

Intab grid cols
`intab_nbc`

- Number of columns of the grid for visualizations in the tab.
- open text
- default: 2
- views_confdef.xml

Dot size
`dot_size`

- Size of the dots in points.
- range in [0.0, 10.0]
- default: 3.0
- views_confdef.xml

Font size
`plot_fontsize`

- Font size used in plots for ticks, etc.
- range in [0, 100]
- default: 12
- views_confdef.xml

Details level
`details_level`

- Level of details for plotting the parallel coordinates.
- range in [0, 100]
- default: 50
- views_confdef.xml

Max emph lbls
`max_emphlbl`

- More that this number of entities selected at once will not get individual labels.
- range in [-1, 20]
- default: 5
- views_confdef.xml

Dot shape

`dot_shape`

- Shape of the dots
- single option in {s, o, p, h, 8, ., ,, *, +, x}
- default: s
- views_confdef.xml

Transparency off
`alpha_off`

- Turn transparency off.
- yes/no
- default: no
- views_confdef.xml

Add stamp
`add_stamp`

- Stamp redescription above the figure.
- yes/no
- default: no
- views_confdef.xml

Detailed missing
`miss_details`

- Show details of different missing values.
- yes/no
- default: yes
- views_confdef.xml

Draw uncovered
`draw_delta`

- Draw uncovered entities (map)
- yes/no
- default: yes
- views_confdef.xml

Click entities
`click_entities`

- Show details entities labels on click.
- yes/no
- default: yes
- views_confdef.xml

Hover entities
`hover_entities`

- Show details entities labels on hover.

- yes/no
- default: no
- views_confdef.xml

Hover coordinates
`hover_coords`

- Show coordinates on hover.
- yes/no
- default: no
- views_confdef.xml

Show literals contribution
`literals_contrib`

- Show contribution of literals (PaCo)
- yes/no
- default: no
- views_confdef.xml

Colors: Colors

Custom cluster colors
`custom_ccmap`

- Custom colors for plotting clusters (matplotlib cmap name, list or dictionnary of hexadecimal colors).
- open text
- default:
- views_confdef.xml

Color LHS
`color_l`

- Color for plotting the LHS support.
- color #RRGGBB
- default: #fc5864
- views_confdef.xml

Color RHS
`color_r`

- Color for plotting the RHS support.
- color #RRGGBB
- default: #74a8f6
- views_confdef.xml

Color Intersection
`color_i`

- Color for plotting the intersection support.

- color #RRGGBB
- default: #662a8d
- views_confdef.xml

Color Uncovered
`color_o`

- Color for plotting the uncovered entities.
- color #RRGGBB
- default: #e6e6e6
- views_confdef.xml

Color Highlight
`color_h`

- Color for highlighting entities.
- color #RRGGBB
- default: #ffff00
- views_confdef.xml

List views: Vizualizations

Max nb. clusters
`max_clus`

- Maximum number of clusters in similarity list view.
- open text
- default: 20
- views_confdef.xml

Values for blocks
`blocks_disp_values`

- Property to use as value for blocks.
- single option in {fractions, percentages, counts}
- default: fractions
- views_confdef.xml

Re-order r.ids for blocks
`blocks_reorder_rids`

- Reorder redescriptions for blocks.
- yes/no
- default: yes
- views_confdef.xml

Show r.ids for blocks
`blocks_show_rids`

- Show redescriptions ids for blocks.

- yes/no
- default: yes
- views_confdef.xml

Show values for blocks
`blocks_show_values`

- Show values for blocks.
- yes/no
- default: no
- views_confdef.xml

Draw selection block
`blocks_show_emph`

- Draw special block for selection.
- yes/no
- default: yes
- views_confdef.xml

Support parts
`supp_part_clus`

- Support parts to use in computing clusters.
- multiple options in {Exx, Exo, Eox, Eoo}
- default: {Exx}
- views_confdef.xml

Missing support parts
`miss_part_clus`

- Missing support parts to use in computing clusters.
- multiple options in {Exm, Emx, Eom, Emo, Emm}
- default: {}
- views_confdef.xml

Maps elems.: Elements

Map background opacity
`map_back_alpha`

- Opacity of the background image.
- range in [0, 100]
- default: 100
- views_confdef.xml

Map background scale
`map_back_scale`

- Scale of the background image.

- range in [0, 100]
- default: 100
- views_confdef.xml

Map background
`map_back`

- Background to use for the map (only with Basemap, not with Cartopy).
- single option in {none, shaded relief, etopo relief, NASA Blue Marble}
- default: none
- views_confdef.xml

Overlay
`map_lines_overlay`

- Overlay map lines on top of other plot elements.
- yes/no
- default: no
- views_confdef.xml

Polygons
`map_poly`

- Use polygons to represent entities on the map
- yes/no
- default: yes
- views_confdef.xml

Map natural areas
`map_elem_area`

- Map natural areas to paint.
- multiple options in {seas, continents, lakes}
- default: {seas}
- views_confdef.xml

Map natural elements
`map_elem_natural`

- Map natural elements to draw.
- multiple options in {coasts, rivers}
- default: {coasts}
- views_confdef.xml

Map geopolitic elements
`map_elem_geop`

- Map geopolitic borders to paint.
- multiple options in {countries, states}
- default: {countries}

- views_confdef.xml

Map imaginary circles
`map_elem_circ`

- Map imaginary circles to draw.
- multiple options in {parallels, meridians}
- default: {}
- views_confdef.xml

Line color
`line_color`

- Color for drawing map lines.
- color #RRGGBB
- default: #808080
- views_confdef.xml

Sea color
`sea_color`

- Color for filling water bodies in maps.
- color #RRGGBB
- default: #f9fcff
- views_confdef.xml

Land color
`land_color`

- Color for filling land areas in maps.
- color #RRGGBB
- default: #ffffff
- views_confdef.xml

Maps proj.: Projection

Left corner longitude
`llon`

- Map boundary's left corner longitude.
- range in [-180.0, 181.0]
- default: 181.0
- views_confdef.xml

Right corner longitude
`ulon`

- Map boundary's right corner longitude.
- range in [-180.0, 181.0]
- default: 181.0

- views_confdef.xml

Lower corner latitude
`llat`

- Map boundary's lower corner latitude.
- range in [-180.0, 181.0]
- default: 181.0
- views_confdef.xml

Upper corner latitude
`ulat`

- Map boundary's upper corner latitude.
- range in [-180.0, 181.0]
- default: 181.0
- views_confdef.xml

Resolution
`map_resolution`

- Resolution of the map elements.
- single option in {crude, low, intermediate, high, full}
- default: crude
- views_confdef.xml

Projection
`map_proj`

- Projection to use for the map.
- single option in {none, Plate Carree, Transverse Mercator, Mercator, Lambert Cylindrical, Lambert Conformal, Lambert Azimuthal Equal Area, Miller, Rotated Pole, Gnomonic, Stereographic, North Polar Stereo, South Polar Stereo, Orthographic, Mollweide, Robinson, Interrupted Goode Homolosine, Geostationary, Albers Equal Area, Azimuthal Equidistant, Sinusoidal}
- default: Miller
- views_confdef.xml

4.3 Other parameters

It is possible to adjust the parameters through the interface then export them to a file. In particular, a template can be generated and exported which can be used to run the *ReReMi* mining algorithm from the command line, after filling in input/output parameters, which are specified here.

Further parameters pertain to randomization, cross-fold tests, server parameters, etc.

File Names: Data Input

LHS data filename
`LHS_data`

- Complete filename for the left hand side data.

- open text
- default:
- inout_confdef.xml

RHS data filename
`RHS_data`

- Complete filename for the right hand side data.
- open text
- default:
- inout_confdef.xml

Data repertory
`data_rep`

- Repertory where the data files are stored.
- open text
- default: __TMP_DIR__
- inout_confdef.xml

LHS data basename
`data_l`

- Basename of the file containing the left hand side data.
- open text
- default: left
- inout_confdef.xml

RHS data basename
`data_r`

- Basename of the file containing the right hand side data.
- open text
- default: right
- inout_confdef.xml

LHS data extension
`ext_l`

- Extension of the file containing the left hand side data.
- open text
- default: .csv
- inout_confdef.xml

RHS data extension
`ext_r`

- Extension of the file containing the right hand side data.
- open text
- default: .csv

- inout_confdef.xml

NA representation
`NA_str`

- String representation for missing values (CSV format).
- open text
- default: nan
- inout_confdef.xml

Delimiter
`delim_in`

- Field delimiter (CSV format).
- single option in {(auto), TAB, ;, ,, SPC}
- default: (auto)
- inout_confdef.xml

Time year first
`time_yearfirst`

- Whether to interpret the first value in an ambiguous 3-integer date as the year (dateutils).
- single option in {(auto), yes, no}
- default: (auto)
- inout_confdef.xml

Time day first
`time_dayfirst`

- Whether to interpret the first value in an ambiguous 3-integer date as the day (dateutils).
- single option in {(auto), yes, no}
- default: (auto)
- inout_confdef.xml

File Names: Results Output

Results repertory
`result_rep`

- Repertory where to save the results.
- open text
- default: __TMP_DIR__
- inout_confdef.xml

Results data filename
`out_base`

- Name of the file where to store the results.
- open text
- default: results

- inout_confdef.xml

Queries extension
`ext_queries`

- Extension of the file where to save the queries.
- open text
- default: .queries
- inout_confdef.xml

Supports extension
`ext_support`

- Extension of the file where to save the supports.
- open text
- default: .supports
- inout_confdef.xml

Log data extension
`ext_log`

- Extension of the file where to save log traces.
- open text
- default: .txt
- inout_confdef.xml

Queries filename
`queries_file`

- Complete filename for the queries.
- open text
- default:
- inout_confdef.xml

Named queries filename
`queries_named_file`

- Complete filename for the queries with variable names.
- open text
- default:
- inout_confdef.xml

Support filename
`support_file`

- Complete filename for the support.
- open text
- default:
- inout_confdef.xml

Advanced: File Names

Series ID
`series_id`

- Series ID to replace __SID__ in filenames.
- open text
- default:
- inout_confdef.xml

Logfile
`logfile`

- File to log. + means generate filename from queries output file. - means stdout.
- open text
- default: -
- inout_confdef.xml

Pairs store filename
`pairs_store`

- Name of the file where to store initial pairs.
- open text
- default:
- inout_confdef.xml

Alternate queries
`queries_alternate`

- Alternate filename or suffix for the queries, for post-processing.
- open text
- default:
- inout_confdef.xml

Advanced: Extra

Mask for LHS variables
`mask_vars_LHS`

- Mask for enabling or disabling LHS variables, overriding any info in the data file.
- open text
- default:
- inout_confdef.xml

Mask for RHS variables
`mask_vars_RHS`

- Mask for enabling or disabling RHS variables, overriding any info in the data file.
- open text
- default:

- inout_confdef.xml

Mask for rows
`mask_rows`

- Mask for enabling or disabling rows, overriding any info in the data file.
- open text
- default:
- inout_confdef.xml

Extensions repertory
`extensions_rep`

- Repertory where extension data files are stored.
- open text
- default: __TMP_DIR__
- inout_confdef.xml

Extensions data filenames
`extensions_names`

- Keys and names of the extension data files.
- open text
- default:
- inout_confdef.xml

Files defining reds fields
`fields_rdefs`

- Files defining extra redescriptions fields.
- open text
- default:
- inout_confdef.xml

Files defining vars fields
`fields_vdefs`

- Files defining extra variables fields.
- open text
- default:
- inout_confdef.xml

Files defining reds actions
`actions_rdefs`

- Files defining extra redescriptions actions.
- open text
- default:
- inout_confdef.xml

Activated extensions

`activated_extensions`

- Data extensions to activate.
- multiple options in {(auto), geoplus}
- default: {}
- inout_confdef.xml

Network: Work client

Results delay
`results_delay`

- Delay between results checks by the client, in seconds.
- open text
- default: 5
- network_confdef.xml

Network: Work server

Workserver IP
`workserver_ip`

- IP adress of the server executing the computational tasks or local for local computations.
- open text
- default: local
- network_confdef.xml

Workserver Port
`workserver_port`

- Listening port of the server executing the computational tasks.
- open text
- default: 55444
- network_confdef.xml

Workserver Key
`workserver_authkey`

- Authentification key on the server executing the computational tasks.
- open text
- default: sesame
- network_confdef.xml

Client Id
`workserver_clientid`

- Client ID to reconnect on the server.
- open text
- default: 0

- network_confdef.xml

Extensions: geoplus

H grid percentile
`hgrid_percentile`

- Percentile to determine grid height.
- range in [0.0, 100.0]
- default: 90.0
- dataext_confdef.xml

W grid percentile
`wgrid_percentile`

- Factor to multiply grid width.
- range in [0.0, 100.0]
- default: 90.0
- dataext_confdef.xml

Distance type
`dst_type`

- Distance to use when computing the grid
- single option in {globe, flat}
- default: globe
- dataext_confdef.xml

Draw tighter borders
`after_cut`

- Draw tighter border between distant sites.
- yes/no
- default: yes
- dataext_confdef.xml

Folds: Automatic splits

Strips direction
`coo_dim`

- Coordinate to cut the strips along (-1 and -2 for cut along longitudes and latitudes, respectively).
- open text
- default: -1
- folds_confdef.xml

Granularity
`grain`

- Granularity of the cut (lower means narrower strips).

- open text
- default: 1.0
- folds_confdef.xml

Variable
`folds_col`

- Name of the variable to use for splitting the data into subsets.
- open text
- default:
- folds_confdef.xml

Nb folds
`nb_folds`

- Number of parts the data should be split into.
- range in [2, 20]
- default: 5
- folds_confdef.xml

Randomization: Data Input

Traits data filename
`traits_data`

- Complete filename for the traits data.
- open text
- default:
- rnd_confdef.xml

Traits data basename
`data_t`

- Basename of the file containing the traits data.
- open text
- default:
- rnd_confdef.xml

Traits data extension
`ext_t`

- Extension of the file containing the traits data.
- open text
- default: .csv
- rnd_confdef.xml

Name count variable
`count_vname`

- Name for the variable that stores occurrence counts.

- open text
- default: COUNTS
- rnd_confdef.xml

Selection redescription
`select_red`

- Redescription used to select data rows.
- open text
- default:
- rnd_confdef.xml

Agg. precision
`agg_prec`

- Precision of aggregated data.
- open text
- default: -1
- rnd_confdef.xml

Randomization: Series

Randomization serie size
`rnd_series_size`

- Number of runs per randomization series.
- open text
- default: 10
- rnd_confdef.xml

Randomization seed
`rnd_seed`

- Seed for the random number generator.
- open text
- default: -1
- rnd_confdef.xml

Randomization meth.
`rnd_meth`

- Randomization method used to generate data.
- multiple options in {none, permute_RHS, permute_LHS, permute_traits, shuffle_traits, swaprnd_occs}
- default: {none}
- rnd_confdef.xml

INDEX

Symbols

(auto)
  activated_extensions, 50
  delim_in, 47
  time_dayfirst, 47
  time_yearfirst, 47
color #RRGGBB parameters
  color_h, 41
  color_i, 40
  color_l, 40
  color_o, 41
  color_r, 40
  land_color, 44
  line_color, 44
  sea_color, 44
multiple options parameters
  activated_extensions, 50
  inits_types_exclude, 29
  map_elem_area, 43
  map_elem_circ, 44
  map_elem_geop, 43
  map_elem_natural, 43
  miss_part_clus, 42
  neg_query_s0_1, 33
  neg_query_s0_2, 34
  neg_query_s0_3, 34
  neg_query_s1_1, 34
  neg_query_s1_2, 34
  neg_query_s1_3, 34
  ops_query_s0, 34
  ops_query_s1, 34
  rnd_meth, 54
  supp_part_clus, 42
open text parameters
  actions_rdefs, 50
  agg_prec, 54
  batch_cap, 36
  batch_out, 31
  coo_dim, 52
  count_vname, 53
  custom_ccmap, 40
  data_l, 46
  data_r, 46
  data_rep, 46
  data_t, 53
  ext_l, 46
  ext_log, 48
  ext_queries, 48
  ext_r, 46
  ext_support, 48
  ext_t, 53
  extensions_names, 50
  extensions_rep, 50
  fields_rdefs, 50
  fields_vdefs, 50
  folds_col, 53
  grain, 52
  in_weight, 31
  init_minscore, 35
  intab_nbc, 38
  intab_nbr, 38
  LHS_data, 45
  logfile, 49
  lower_tail_agg, 37
  mask_rows, 50
  mask_vars_LHS, 49
  mask_vars_RHS, 49
  max_agg, 37
  max_clus, 41
  max_fin_pval, 30
  max_inits, 28
  max_overlap_area, 30
  max_overlap_rows, 30
  max_prodbuckets, 37
  max_rounds, 32
  max_seg, 37
  max_var_s0, 27
  max_var_s1, 28
  min_fin_acc, 29
  min_fin_in, 29
  min_fin_out, 29
  min_fin_var, 31
  min_impr, 28
  min_itm_c, 28

min_itm_in, 28
min_itm_out, 28
NA_str, 47
nb_processes, 30
out_base, 47
out_weight, 31
pairs_store, 49
queries_alternate, 49
queries_file, 48
queries_named_file, 48
result_rep, 47
results_delay, 51
RHS_data, 46
rnd_seed, 54
rnd_series_size, 54
score.impacc, 35
score.pval_fact, 36
score.pval_query, 36
score.pval_red, 36
score.rel_impacc, 35
select_red, 54
series_id, 49
splittrees_init_depth, 32
support_file, 48
traits_data, 53
upper_tail_agg, 37
workserver_authkey, 51
workserver_clientid, 51
workserver_ip, 51
workserver_port, 51

parameters defined in dataext_confdef.xml
after_cut, 52
dst_type, 52
hgrid_percentile, 52
wgrid_percentile, 52

parameters defined in folds_confdef.xml
coo_dim, 52
folds_col, 53
grain, 52
nb_folds, 53

parameters defined in inout_confdef.xml
actions_rdefs, 50
activated_extensions, 50
data_l, 46
data_r, 46
data_rep, 46
delim_in, 47
ext_l, 46
ext_log, 48
ext_queries, 48
ext_r, 46
ext_support, 48
extensions_names, 50
extensions_rep, 50
fields_rdefs, 50
fields_vdefs, 50
LHS_data, 45
logfile, 49
mask_rows, 50
mask_vars_LHS, 49
mask_vars_RHS, 49
NA_str, 47
out_base, 47
pairs_store, 49
queries_alternate, 49
queries_file, 48
queries_named_file, 48
result_rep, 47
RHS_data, 46
series_id, 49
support_file, 48
time_dayfirst, 47
time_yearfirst, 47

parameters defined in miner_confdef.xml
add_condition, 33
amnesic, 36
batch_cap, 36
batch_out, 31
debug_checks, 30
in_weight, 31
init_minscore, 35
inits_productivity, 29
inits_types_exclude, 29
lower_tail_agg, 37
max_agg, 37
max_fin_pval, 30
max_inits, 28
max_overlap_area, 30
max_overlap_rows, 30
max_prodbuckets, 37
max_rounds, 32
max_seg, 37
max_var_s0, 27
max_var_s1, 28
method_pval, 31
min_fin_acc, 29
min_fin_in, 29
min_fin_out, 29
min_fin_var, 31
min_impr, 28
min_itm_c, 28
min_itm_in, 28
min_itm_out, 28
mining_algo, 27

multi_cats, 33
nb_processes, 30
neg_query_s0_1, 33
neg_query_s0_2, 34
neg_query_s0_3, 34
neg_query_s1_1, 34
neg_query_s1_2, 34
neg_query_s1_3, 34
ops_query_s0, 34
ops_query_s1, 34
out_weight, 31
pair_sel, 35
parts_type, 32
pe_balance, 35
score.impacc, 35
score.pval_fact, 36
score.pval_query, 36
score.pval_red, 36
score.rel_impacc, 35
single_side_or, 33
split_criterion, 32
splittrees_both_sides_change, 33
splittrees_dyn_side, 33
splittrees_init_depth, 32
splittrees_update_target, 32
upper_tail_agg, 37
var_compat, 31
verbosity, 30

parameters defined in network_confdef.xml
results_delay, 51
workserver_authkey, 51
workserver_clientid, 51
workserver_ip, 51
workserver_port, 51

parameters defined in rnd_confdef.xml
agg_prec, 54
count_vname, 53
data_t, 53
ext_t, 53
rnd_meth, 54
rnd_seed, 54
rnd_series_size, 54
select_red, 54
traits_data, 53

parameters defined in views_confdef.xml
add_stamp, 39
alpha_off, 39
blocks_disp_values, 41
blocks_reorder_rids, 41
blocks_show_emph, 42
blocks_show_rids, 41
blocks_show_values, 42
click_entities, 39
color_h, 41
color_i, 40
color_l, 40
color_o, 41
color_r, 40
custom_ccmap, 40
details_level, 38
dot_shape, 38
dot_size, 38
draw_delta, 39
hover_coords, 40
hover_entities, 39
intab_nbc, 38
intab_nbr, 38
land_color, 44
line_color, 44
literals_contrib, 40
llat, 45
llon, 44
map_back, 43
map_back_alpha, 42
map_back_scale, 42
map_elem_area, 43
map_elem_circ, 44
map_elem_geop, 43
map_elem_natural, 43
map_lines_overlay, 43
map_poly, 43
map_proj, 45
map_resolution, 45
max_clus, 41
max_emphlbl, 38
miss_details, 39
miss_part_clus, 42
plot_fontsize, 38
sea_color, 44
supp_part_clus, 42
ulat, 45
ulon, 44

range parameters
details_level, 38
dot_size, 38
hgrid_percentile, 52
llat, 45
llon, 44
map_back_alpha, 42
map_back_scale, 42
max_emphlbl, 38
nb_folds, 53
pe_balance, 35
plot_fontsize, 38
ulat, 45

- ulon, 44
- verbosity, 30
- wgrid_percentile, 52

- # single option parameters
  - blocks_disp_values, 41
  - delim_in, 47
  - dot_shape, 38
  - dst_type, 52
  - inits_productivity, 29
  - map_back, 43
  - map_proj, 45
  - map_resolution, 45
  - method_pval, 31
  - mining_algo, 27
  - pair_sel, 35
  - parts_type, 32
  - split_criterion, 32
  - time_dayfirst, 47
  - time_yearfirst, 47
  - var_compat, 31
- # yes/no parameters
  - add_condition, 33
  - add_stamp, 39
  - after_cut, 52
  - alpha_off, 39
  - amnesic, 36
  - blocks_reorder_rids, 41
  - blocks_show_emph, 42
  - blocks_show_rids, 41
  - blocks_show_values, 42
  - click_entities, 39
  - debug_checks, 30
  - draw_delta, 39
  - hover_coords, 40
  - hover_entities, 39
  - literals_contrib, 40
  - map_lines_overlay, 43
  - map_poly, 43
  - miss_details, 39
  - multi_cats, 33
  - single_side_or, 33
  - splittrees_both_sides_change, 33
  - splittrees_dyn_side, 33
  - splittrees_update_target, 32

A

- Accuracy improvement coefficient, 35
- actions_rdefs, 50
- Activated extensions, 50
- activated_extensions, 50
- Add stamp, 39
- add_condition, 33
- add_stamp, 39
- Additional filtering
  - max_overlap_area, 30
  - max_overlap_rows, 30
- Advanced
  - Extra; actions_rdefs, 50
  - Extra; activated_extensions, 50
  - Extra; extensions_names, 50
  - Extra; extensions_rep, 50
  - Extra; fields_rdefs, 50
  - Extra; fields_vdefs, 50
  - Extra; mask_rows, 50
  - Extra; mask_vars_LHS, 49
  - Extra; mask_vars_RHS, 49
  - File Names; logfile, 49
  - File Names; pairs_store, 49
  - File Names; queries_alternate, 49
  - File Names; series_id, 49
- after_cut, 52
- Agg. precision, 54
- agg_prec, 54
- Albers Equal Area
  - map_proj, 45
- alpha_off, 39
- alternate
  - pair_sel, 35
- Alternate queries, 49
- Amnesic, 36
- amnesic, 36
- Automatic splits
  - coo_dim, 52
  - folds_col, 53
  - grain, 52
  - nb_folds, 53
- Azimuthal Equidistant
  - map_proj, 45

B

- basic
  - parts_type, 32
- Batch capacity, 36
- Batch output, 31
- batch_cap, 36
- batch_out, 31
- BB
  - inits_types_exclude, 29
- BC
  - inits_types_exclude, 29
- blocks_disp_values, 41
- blocks_reorder_rids, 41
- blocks_show_emph, 42
- blocks_show_rids, 41
- blocks_show_values, 42
- BN
  - inits_types_exclude, 29
- Both sides change, 33

C

Candidate expansion
    amnesic, 36
    batch_cap, 36
    score.impacc, 35
    score.pval_fact, 36
    score.pval_query, 36
    score.pval_red, 36
    score.rel_impacc, 35
Candidate filtering
    inits_productivity, 29
    inits_types_exclude, 29
    max_inits, 28
    min_impr, 28
    min_itm_c, 28
    min_itm_in, 28
    min_itm_out, 28
cartwheel
    mining_algo, 27
CC
    inits_types_exclude, 29
Click entities, 39
click_entities, 39
Client Id, 51
CN
    inits_types_exclude, 29
coasts
    map_elem_natural, 43
Color Highlight, 41
Color Intersection, 40
Color LHS, 40
Color RHS, 40
Color Uncovered, 41
color_h, 41
color_i, 40
color_l, 40
color_o, 41
color_r, 40
Colors
    color_h, 41
    color_i, 40
    color_l, 40
    color_o, 41
    color_r, 40
    Colors; color_h, 41
    Colors; color_i, 40
    Colors; color_l, 40
    Colors; color_o, 41
    Colors; color_r, 40
    Colors; custom_ccmap, 40
    custom_ccmap, 40
Conditional, 33
conjunction
    ops_query_s0, 34
    ops_query_s1, 34
continents
    map_elem_area, 43
coo_dim, 52
count_vname, 53
countries
    map_elem_geop, 43
counts
    blocks_disp_values, 41
crude
    map_resolution, 45
Custom cluster colors, 40
custom_ccmap, 40

D

Data Input
    agg_prec, 54
    count_vname, 53
    data_l, 46
    data_r, 46
    data_rep, 46
    data_t, 53
    delim_in, 47
    ext_l, 46
    ext_r, 46
    ext_t, 53
    LHS_data, 45
    NA_str, 47
    RHS_data, 46
    select_red, 54
    time_dayfirst, 47
    time_yearfirst, 47
    traits_data, 53
Data repertory, 46
data_l, 46
data_r, 46
data_rep, 46
data_t, 53
Debug checks, 30
debug_checks, 30
delim_in, 47
Delimiter, 47
Detailed missing, 39
Details level, 38
details_level, 38
differ
    var_compat, 31
disjunction
    ops_query_s0, 34
    ops_query_s1, 34
Distance type, 52
Dot shape, 38
Dot size, 38
dot_shape, 38

dot_size, 38
Draw selection block, 42
Draw tighter borders, 52
Draw uncovered, 39
draw_delta, 39
dst_type, 52
Dynamic side, 33

E

Elements
  land_color, 44
  line_color, 44
  map_back, 43
  map_back_alpha, 42
  map_back_scale, 42
  map_elem_area, 43
  map_elem_circ, 44
  map_elem_geop, 43
  map_elem_natural, 43
  map_lines_overlay, 43
  map_poly, 43
  sea_color, 44
Emm
  miss_part_clus, 42
Emo
  miss_part_clus, 42
Emx
  miss_part_clus, 42
entropy
  split_criterion, 32
Eom
  miss_part_clus, 42
Eoo
  supp_part_clus, 42
Eox
  supp_part_clus, 42
etopo relief
  map_back, 43
exclu
  parts_type, 32
Excluded initial cand. pair types, 29
Exm
  miss_part_clus, 42
Exo
  supp_part_clus, 42
Expansion order, 35
ext_l, 46
ext_log, 48
ext_queries, 48
ext_r, 46
ext_support, 48
ext_t, 53
Extensions
  geoplus; after_cut, 52
  geoplus; dst_type, 52
  geoplus; hgrid_percentile, 52
  geoplus; wgrid_percentile, 52
Extensions data filenames, 50
Extensions repertory, 50
extensions_names, 50
extensions_rep, 50
Extra
  actions_rdefs, 50
  activated_extensions, 50
  extensions_names, 50
  extensions_rep, 50
  fields_rdefs, 50
  fields_vdefs, 50
  mask_rows, 50
  mask_vars_LHS, 49
  mask_vars_RHS, 49
Exx
  supp_part_clus, 42

F

fields_rdefs, 50
fields_vdefs, 50
File Names
  Data Input; data_l, 46
  Data Input; data_r, 46
  Data Input; data_rep, 46
  Data Input; delim_in, 47
  Data Input; ext_l, 46
  Data Input; ext_r, 46
  Data Input; LHS_data, 45
  Data Input; NA_str, 47
  Data Input; RHS_data, 46
  Data Input; time_dayfirst, 47
  Data Input; time_yearfirst, 47
  logfile, 49
  pairs_store, 49
  queries_alternate, 49
  Results Output; ext_log, 48
  Results Output; ext_queries, 48
  Results Output; ext_support, 48
  Results Output; out_base, 47
  Results Output; queries_file, 48
  Results Output; queries_named_file, 48
  Results Output; result_rep, 47
  Results Output; support_file, 48
  series_id, 49
Files defining reds actions, 50
Files defining reds fields, 50
Files defining vars fields, 50
fim
  mining_algo, 27
Final filtering

- max_fin_pval, 30
- min_fin_acc, 29
- min_fin_in, 29
- min_fin_out, 29

flat
- dst_type, 52

Folds
- Automatic splits; coo_dim, 52
- Automatic splits; folds_col, 53
- Automatic splits; grain, 52
- Automatic splits; nb_folds, 53

folds_col, 53
Font size, 38
fractions
- blocks_disp_values, 41

full
- map_resolution, 45

G

geoplus
- activated_extensions, 50
- after_cut, 52
- dst_type, 52
- hgrid_percentile, 52
- wgrid_percentile, 52

Geostationary
- map_proj, 45

gini
- split_criterion, 32

globe
- dst_type, 52

Gnomonic
- map_proj, 45

grain, 52
Granularity, 52

H

H grid percentile, 52
hgrid_percentile, 52
high
- inits_productivity, 29
- map_resolution, 45

Hover coordinates, 40
Hover entities, 39
hover_coords, 40
hover_entities, 39

I

in_weight, 31
Init depth, 32
Init min score, 35
init_minscore, 35
Initial candidates productivity, 29
Initial pairs
- init_minscore, 35
- pair_sel, 35
- pe_balance, 35

inits_productivity, 29
inits_types_exclude, 29
Intab grid cols, 38
Intab grid rows, 38
intab_nbc, 38
intab_nbr, 38
intermediate
- map_resolution, 45

Interrupted Goode Homolosine
- map_proj, 45

L

lakes
- map_elem_area, 43

Lambert Azimuthal Equal Area
- map_proj, 45

Lambert Conformal
- map_proj, 45

Lambert Cylindrical
- map_proj, 45

Land color, 44
land_color, 44
layeredtrees
- mining_algo, 27

Left corner longitude, 44
LHS data basename, 46
LHS data extension, 46
LHS data filename, 45
LHS Literals Bool, 33
LHS Literals Cat, 34
LHS Literals Num, 34
LHS max number of variables, 27
LHS Operators, 34
LHS_data, 45
Line color, 44
line_color, 44
List views
- Vizualizations; blocks_disp_values, 41
- Vizualizations; blocks_reorder_rids, 41
- Vizualizations; blocks_show_emph, 42
- Vizualizations; blocks_show_rids, 41
- Vizualizations; blocks_show_values, 42
- Vizualizations; max_clus, 41
- Vizualizations; miss_part_clus, 42
- Vizualizations; supp_part_clus, 42

literals_contrib, 40
llat, 45
llon, 44

Log data extension, 48
Log verbosity, 30
Logfile, 49
logfile, 49
low
    inits_productivity, 29
    map_resolution, 45
Lower corner latitude, 45
Lower tail aggregate, 37
lower_tail_agg, 37

M

Map background, 43
Map background opacity, 42
Map background scale, 42
Map geopolitic elements, 43
Map imaginary circles, 44
Map natural areas, 43
Map natural elements, 43
map_back, 43
map_back_alpha, 42
map_back_scale, 42
map_elem_area, 43
map_elem_circ, 44
map_elem_geop, 43
map_elem_natural, 43
map_lines_overlay, 43
map_poly, 43
map_proj, 45
map_resolution, 45
Maps elems.
    Elements; land_color, 44
    Elements; line_color, 44
    Elements; map_back, 43
    Elements; map_back_alpha, 42
    Elements; map_back_scale, 42
    Elements; map_elem_area, 43
    Elements; map_elem_circ, 44
    Elements; map_elem_geop, 43
    Elements; map_elem_natural, 43
    Elements; map_lines_overlay, 43
    Elements; map_poly, 43
    Elements; sea_color, 44
Maps proj.
    Projection; llat, 45
    Projection; llon, 44
    Projection; map_proj, 45
    Projection; map_resolution, 45
    Projection; ulat, 45
    Projection; ulon, 44
marg
    method_pval, 31
Mask for LHS variables, 49
Mask for RHS variables, 49
Mask for rows, 50
mask_rows, 50
mask_vars_LHS, 49
mask_vars_RHS, 49
match
    var_compat, 31
Max aggregation, 37
Max Area Overlap, 30
Max emph lbls, 38
Max initial candidates, 28
Max nb. clusters, 41
Max number segments, 37
Max p-value, 30
Max product buckets, 37
Max rounds in CartWheel, 32
Max Rows Overlap, 30
max_agg, 37
max_clus, 41
max_emphlbl, 38
max_fin_pval, 30
max_inits, 28
max_overlap_area, 30
max_overlap_rows, 30
max_prodbuckets, 37
max_rounds, 32
max_seg, 37
max_var_s0, 27
max_var_s1, 28
medium
    inits_productivity, 29
Mercator
    map_proj, 45
meridians
    map_elem_circ, 44
Method
    mining_algo, 27
method_pval, 31
Miller
    map_proj, 45
Min accuracy, 29
Min contribution, 28
Min improvement, 28
Min support, 28, 29
Min total number of variables, 31
Min uncovered, 28, 29
min_fin_acc, 29
min_fin_in, 29
min_fin_out, 29
min_fin_var, 31
min_impr, 28
min_itm_c, 28
min_itm_in, 28
min_itm_out, 28
Mining

- Additional filtering; max_overlap_area, 30
- Additional filtering; max_overlap_rows, 30
- Candidate filtering; inits_productivity, 29
- Candidate filtering; inits_types_exclude, 29
- Candidate filtering; max_inits, 28
- Candidate filtering; min_impr, 28
- Candidate filtering; min_itm_c, 28
- Candidate filtering; min_itm_in, 28
- Candidate filtering; min_itm_out, 28
- Final filtering; max_fin_pval, 30
- Final filtering; min_fin_acc, 29
- Final filtering; min_fin_in, 29
- Final filtering; min_fin_out, 29
- Method; mining_algo, 27
- Queries; max_var_s0, 27
- Queries; max_var_s1, 28

Mining adv.
- Other common parameters; batch_out, 31
- Other common parameters; in_weight, 31
- Other common parameters; method_pval, 31
- Other common parameters; min_fin_var, 31
- Other common parameters; out_weight, 31
- Other common parameters; parts_type, 32
- Other common parameters; var_compat, 31
- Process; debug_checks, 30
- Process; nb_processes, 30
- Process; verbosity, 30
- Splittrees variant; splittrees_both_sides_change, 33
- Splittrees variant; splittrees_dyn_side, 33
- Splittrees variant; splittrees_init_depth, 32
- Splittrees variant; splittrees_update_target, 32
- Tree-based algorithms; max_rounds, 32
- Tree-based algorithms; split_criterion, 32

Mining algorithm, 27
mining_algo, 27
miss_details, 39
miss_part_clus, 42
Missing support parts, 42
Missing values belief, 32
Mollweide
- map_proj, 45

Multi categories, 33
multi_cats, 33

N

NA representation, 47
NA_str, 47
Name count variable, 53
Named queries filename, 48
NASA Blue Marble
- map_back, 43

Nb folds, 53
Nb processes, 30
nb_folds, 53
nb_processes, 30
neg_query_s0_1, 33
neg_query_s0_2, 34
neg_query_s0_3, 34
neg_query_s1_1, 34
neg_query_s1_2, 34
neg_query_s1_3, 34
negative
- neg_query_s0_1, 33
- neg_query_s0_2, 34
- neg_query_s0_3, 34
- neg_query_s1_1, 34
- neg_query_s1_2, 34
- neg_query_s1_3, 34
- parts_type, 32

Network
- Work client; results_delay, 51
- Work server; workserver_authkey, 51
- Work server; workserver_clientid, 51
- Work server; workserver_ip, 51
- Work server; workserver_port, 51

NN
- inits_types_exclude, 29

no
- time_dayfirst, 47
- time_yearfirst, 47

none
- map_back, 43
- map_proj, 45
- rnd_meth, 54

North Polar Stereo
- map_proj, 45

Numerical attributes discretization
- lower_tail_agg, 37
- max_agg, 37
- max_prodbuckets, 37

    max_seg, 37
    upper_tail_agg, 37

O

ops_query_s0, 34
ops_query_s1, 34
optimistic
    parts_type, 32
Orthographic
    map_proj, 45
Other common parameters
    batch_out, 31
    in_weight, 31
    method_pval, 31
    min_fin_var, 31
    out_weight, 31
    parts_type, 32
    var_compat, 31
out_base, 47
out_weight, 31
over
    method_pval, 31
overall
    pair_sel, 35
Overlay, 43

P

p-value factor, 36
p-value formula, 31
pair_sel, 35
Pairs store filename, 49
Pairs-expand balance, 35
pairs_store, 49
parallels
    map_elem_circ, 44
parts_type, 32
pe_balance, 35
percentages
    blocks_disp_values, 41
permute_LHS
    rnd_meth, 54
permute_RHS
    rnd_meth, 54
permute_traits
    rnd_meth, 54
pessimistic
    parts_type, 32
Plate Carree
    map_proj, 45
plot_fontsize, 38
Polygons, 43
positive
    neg_query_s0_1, 33
    neg_query_s0_2, 34
    neg_query_s0_3, 34
    neg_query_s1_1, 34
    neg_query_s1_2, 34
    neg_query_s1_3, 34
    parts_type, 32
Process
    debug_checks, 30
    nb_processes, 30
    verbosity, 30
Projection, 45
    llat, 45
    llon, 44
    map_proj, 45
    map_resolution, 45
    ulat, 45
    ulon, 44

Q

Queries
    add_condition, 33
    max_var_s0, 27
    max_var_s1, 28
    multi_cats, 33
    neg_query_s0_1, 33
    neg_query_s0_2, 34
    neg_query_s0_3, 34
    neg_query_s1_1, 34
    neg_query_s1_2, 34
    neg_query_s1_3, 34
    ops_query_s0, 34
    ops_query_s1, 34
    single_side_or, 33
Queries extension, 48
Queries filename, 48
queries_alternate, 49
queries_file, 48
queries_named_file, 48
Query p-value coefficient, 36

R

Randomization
    Data Input; agg_prec, 54
    Data Input; count_vname, 53
    Data Input; data_t, 53
    Data Input; ext_t, 53
    Data Input; select_red, 54
    Data Input; traits_data, 53
    Series; rnd_meth, 54
    Series; rnd_seed, 54
    Series; rnd_series_size, 54
Randomization meth., 54
Randomization seed, 54
Randomization serie size, 54
Re-order r.ids for blocks, 41

Redescription p-value coefficient, 36
rejective
    parts_type, 32
Relative Accurracy improvement coefficient, 35
ReReMi
    Initial pairs; init_minscore, 35
    Initial pairs; pair_sel, 35
    Initial pairs; pe_balance, 35
    Queries; add_condition, 33
    Queries; multi_cats, 33
    Queries; neg_query_s0_1, 33
    Queries; neg_query_s0_2, 34
    Queries; neg_query_s0_3, 34
    Queries; neg_query_s1_1, 34
    Queries; neg_query_s1_2, 34
    Queries; neg_query_s1_3, 34
    Queries; ops_query_s0, 34
    Queries; ops_query_s1, 34
    Queries; single_side_or, 33
reremi
    mining_algo, 27
ReReMi exp.
    Candidate expansion; amnesic, 36
    Candidate expansion; batch_cap, 36
    Candidate expansion; score.impacc, 35
    Candidate expansion; score.pval_fact, 36
    Candidate expansion; score.pval_query, 36
    Candidate expansion; score.pval_red, 36
    Candidate expansion; score.rel_impacc, 35
    Numerical attributes discretization; lower_tail_agg, 37
    Numerical attributes discretization; max_agg, 37
    Numerical attributes discretization; max_prodbuckets, 37
    Numerical attributes discretization; max_seg, 37
    Numerical attributes discretization; upper_tail_agg, 37
Resolution, 45
result_rep, 47
Results data filename, 47
Results delay, 51
Results Output
    ext_log, 48
    ext_queries, 48
    ext_support, 48
    out_base, 47
    queries_file, 48
    queries_named_file, 48
    result_rep, 47
    support_file, 48
Results repertory, 47
results_delay, 51
RHS data basename, 46
RHS data extension, 46
RHS data filename, 46
RHS Literals Bool, 34
RHS Literals Cat, 34
RHS Literals Num, 34
RHS max number of variables, 28
RHS Operators, 34
RHS_data, 46
Right corner longitude, 44
rivers
    map_elem_natural, 43
rnd_meth, 54
rnd_seed, 54
rnd_series_size, 54
Robinson
    map_proj, 45
Rotated Pole
    map_proj, 45

S

score.impacc, 35
score.pval_fact, 36
score.pval_query, 36
score.pval_red, 36
score.rel_impacc, 35
Sea color, 44
sea_color, 44
seas
    map_elem_area, 43
select_red, 54
Selection redescription, 54
Series
    rnd_meth, 54
    rnd_seed, 54
    rnd_series_size, 54
Series ID, 49
series_id, 49
shaded relief
    map_back, 43
Show literals contribution, 40
Show r.ids for blocks, 41
Show values for blocks, 42
shuffle_traits
    rnd_meth, 54

Single side OR, 33
single_side_or, 33
Sinusoidal
  map_proj, 45
South Polar Stereo
  map_proj, 45
SPC
  delim_in, 47
Split criterion, 32
split_criterion, 32
splittrees
  mining_algo, 27
Splittrees variant
  splittrees_both_sides_change, 33
  splittrees_dyn_side, 33
  splittrees_init_depth, 32
  splittrees_update_target, 32
splittrees_both_sides_change, 33
splittrees_dyn_side, 33
splittrees_init_depth, 32
splittrees_update_target, 32
states
  map_elem_geop, 43
Stereographic
  map_proj, 45
Strips direction, 52
supp
  method_pval, 31
supp_part_clus, 42
Support filename, 48
Support parts, 42
support_file, 48
Supports extension, 48
swaprnd_occs
  rnd_meth, 54

T

TAB
  delim_in, 47
Time day first, 47
Time year first, 47
time_dayfirst, 47
time_yearfirst, 47
Traits data basename, 53
Traits data extension, 53
Traits data filename, 53
traits_data, 53
Transparency off, 39
Transverse Mercator
  map_proj, 45
Tree-based algorithms
  max_rounds, 32
  split_criterion, 32

U

ulat, 45
ulon, 44
Update target, 32
Upper corner latitude, 45
Upper tail aggregate, 37
upper_tail_agg, 37

V

Values for blocks, 41
Var group compatibility, 31
var_compat, 31
Variable, 53
verbosity, 30
Views
  Vizualizations; add_stamp, 39
  Vizualizations; alpha_off, 39
  Vizualizations; click_entities, 39
  Vizualizations; details_level, 38
  Vizualizations; dot_shape, 38
  Vizualizations; dot_size, 38
  Vizualizations; draw_delta, 39
  Vizualizations; hover_coords, 40
  Vizualizations; hover_entities, 39
  Vizualizations; intab_nbc, 38
  Vizualizations; intab_nbr, 38
  Vizualizations; literals_contrib, 40
  Vizualizations; max_emphlbl, 38
  Vizualizations; miss_details, 39
  Vizualizations; plot_fontsize, 38
Vizualizations
  add_stamp, 39
  alpha_off, 39
  blocks_disp_values, 41
  blocks_reorder_rids, 41
  blocks_show_emph, 42
  blocks_show_rids, 41
  blocks_show_values, 42
  click_entities, 39
  details_level, 38
  dot_shape, 38
  dot_size, 38
  draw_delta, 39
  hover_coords, 40
  hover_entities, 39
  intab_nbc, 38
  intab_nbr, 38
  literals_contrib, 40
  max_clus, 41
  max_emphlbl, 38
  miss_details, 39
  miss_part_clus, 42
  plot_fontsize, 38
  supp_part_clus, 42

W

W grid percentile, 52
Weight Inside, 31
Weight Outside, 31
wgrid_percentile, 52
Work client
  results_delay, 51
Work server
  workserver_authkey, 51
  workserver_clientid, 51
  workserver_ip, 51
  workserver_port, 51
Workserver IP, 51
Workserver Key, 51
Workserver Port, 51
workserver_authkey, 51
workserver_clientid, 51
workserver_ip, 51
workserver_port, 51

Y

yes
  time_dayfirst, 47
  time_yearfirst, 47